NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

**SOFTWARE RISK MANAGEMENT:
A CASE STUDY OF THE V-22 PROGRAM**

by

Lloyd R. Whitworth

March 1996

Thesis Advisor: Martin J. McCaffrey

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE			Form Approved OMB No 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 1996	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE SOFTWARE RISK MANAGEMENT: A CASE STUDY OF THE V-22 PROGRAM	5. FUNDING NUMBERS
6. AUTHOR(S) Whitworth, Lloyd R.	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

Over the past thirty years, software development has become an increasingly important part of the technologically advanced weapon systems acquired by DOD. Program offices for software intensive weapon systems are facing the difficult task of managing software development risk. The purpose of this thesis is to identify and analyze software risk management techniques for their general application to software management problems during the acquisition process. This thesis focused on software risk management and risk management techniques used by the V-22 program office. Lessons learned which can be applicable to other programs are identified. The principal finding is that a formal, systematic, and disciplined risk management process, which includes software risk management, must be in place for software intensive weapon system acquisitions. Two primary recommendations are that the program manager create an environment where risks are freely communicated and that program executive officers assist program managers in the identification of software related development risks by conducting independent assessments.

14. SUBJECT TERMS Software Risk Management, Risk Management, V-22 Weapon System			15. NUMBER OF PAGES 147
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500 Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std. 239-18 298-102

SOFTWARE RISK MANAGEMENT:
A CASE STUDY OF THE V-22 PROGRAM

Lloyd R. Whitworth
Major, United States Marine Corps
B.B.A., University of Texas at San Antonio, 1981

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
March 1996

ABSTRACT

Over the past thirty years, software development has become an increasingly important part of the technologically advanced weapon systems acquired by DOD. Program offices for software intensive weapon systems are facing the difficult task of managing software development risk. The purpose of this thesis is to identify and analyze software risk management techniques for their general application to software management problems during the acquisition process. This thesis focused on software risk management and risk management techniques used by the V-22 program office. Lessons learned which can be applicable to other programs are identified. The principal finding is that a formal, systematic, and disciplined risk management process, which includes software risk management, must be in place for software intensive weapon system acquisitions. Two primary recommendations are that the program manager create an environment where risks are freely communicated and that program executive officers assist program managers in the identification of software related development risks by conducting independent assessments.

TABLE OF CONTENTS

I. INTRODUCTION

A. THE SOFTWARE RISK MANAGEMENT CHALLENGE

Over the past thirty years, software has become an increasingly important part of the technologically advanced weapon systems acquired by the Department of Defense (DOD). Management of the software development process is extremely challenging and has become a major source of problems in the system acquisition field [Ref 1:p. 1]. These problems have manifested themselves in a variety of critical weapon systems ranging from submarines to transport aircraft.

Program offices for software intense weapon systems are facing the difficult task of managing software development risk. Managing this risk involves identifying, addressing, and eliminating software risk items before they become either threats to successful operation or major sources of software rework [Ref. 2:p. 1]. This study will document the process of one program office in applying software risk management techniques. Concepts, strategies, and techniques for software risk management can be captured from this case for use in future work.

B. OBJECTIVES

This thesis will examine software risk management in the V-22 aircraft acquisition program of the Department of the Navy. The specific software risk management techniques for this program will be identified and examined. These techniques will then be analyzed for their general application to software management problems during the acquisition process. The overall objective of this thesis is to document the process that will

assist in developing successful strategies for identifying, addressing, and eliminating software risks.

C. SCOPE

This thesis will focus on software risk management techniques that are being used by the program office during the Engineering and Manufacturing Development (E&MD) phase of the acquisition process. Technical aspects of the V-22 program may be presented as they apply to managerial decisions. The central focus will be the challenges facing the program office from the beginning of E&MD to the present time. Full-Scale Development (FSD) issues will be discussed to the extent necessary to explain actions during E&MD.

This research focuses on those computer software configuration items (CSCIs) that best illustrate the application and results of software risk management techniques. The avionics and flight control system software are the CSCIs that present the greatest challenge with respect to software risk management in the E&MD phase.

D. METHODOLOGY

This research is conducted as a case study. A literature review of information pertaining to software risk management was conducted. This review provided the necessary background from which to begin the case analysis.

Analysis of the case required gathering information on the V-22 program weapon system acquisition. A specific focus was with regard to software development and software management of the program for the avionics and flight control system software. This information was gathered through on site interviews with key Government and contractor personnel working directly on the V-22 program. Numerous V-22 General

Accounting Office (GAO) reports, DOD Inspector General reports, and pertinent program documents from the program office were also reviewed.

E. ORGANIZATION

Chapter II establishes the background for the study by discussing the important role of software in the DOD acquisition process. The chapter also defines risk and risk management and provides DOD policy guidance on risk management. Software risk management is defined, and the evolution of software risk management is discussed. The relationship of software risk management to the DOD acquisition process is described.

Chapter III introduces the V-22 weapon system and briefly details the acquisition history of the program. The chapter provides an overview of the major CSCIs on the V-22. It also discusses the transition from FSD to E&MD along with the important role of independent risk assessment teams (IRATs).

Chapter IV describes the V-22 risk management philosophy and risk management process for the V-22 weapon system. It then discusses implementation of software risk management and various factors that have affected the software risk management process. It also explains key Government and contractor actions with respect to managing software risks in the E&MD phase of the acquisition process. Chapter IV finishes by providing an example of risk management.

Chapter V provides an analysis of the factors that have had a significant impact on software risk management in the V-22 program. Lessons learned that can be applied to other programs will also be identified in this chapter.

Chapter VI will provide the conclusion. It will also provide a set of recommendations related to the lessons learned identified in Chapter V.

II. BACKGROUND

A. INTRODUCTION

To better understand the need for software risk management it is necessary to first understand the growing importance of software in DOD weapon systems. This chapter begins by discussing the important role software plays in DOD weapon systems. Next, some DOD software development problems will be identified. Risk and risk management will then be described, as well as the DOD policy and guidance on risk management. Software risk management will then be defined, followed by a discussion on the evolution of software risk management. Finally, the importance of software risk management to the DOD acquisition process will be made.

B. THE ROLE OF SOFTWARE IN DOD WEAPON SYSTEMS

In 1987, the "Report of the Defense Science Board Task Force on Military Software" described the role of military software in this way:

> Software plays a major role in today's weapon systems. The "smarts" of smart weapons are provided by software. Software is crucial to intelligence, communications, command, and control.[...]Software provides a major component of U.S. war-fighting capability.[Ref. 3]

The use of embedded software provides the ability to change or increase the functionality and capabilities of a weapon system, often with little or no effect on hardware characteristics. Software performs many of the critical functions in key weapon systems that cannot be performed by hardware alone. In essence, our key weapon systems today are completely dependent upon software to function properly.[Ref. 1:p. 7]

1. Software Size, Growth, and Complexity

An objective of the U.S. National Defense Strategy is to maintain technological superiority in weapon systems [Ref. 4]. The "high-tech" weapons that have evolved under this strategy during the last three decades have seen an exponential growth in software costs as a percentage of total computer resources [Ref. 1:pp. 7-8].

The growth in software cost has primarily been a result of the growth in volume and complexity of software demanded by DOD. As seen by the chart in Figure 1, below, the growth in software in just the last 10 years has been tremendous.

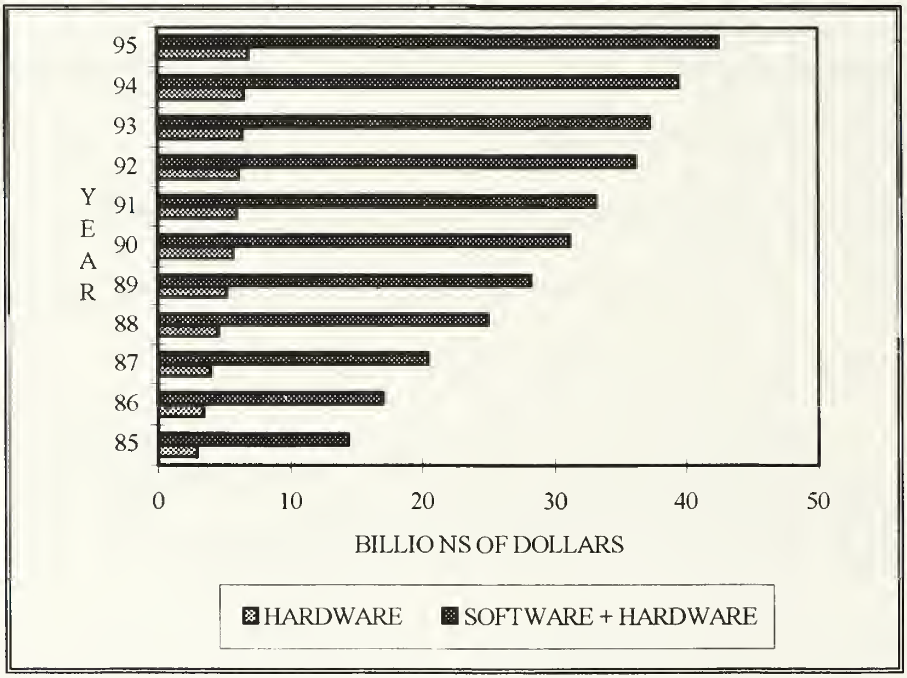


Figure 1. Growth In DOD Embedded Computer Market. From Ref. [5]

As weapon systems have become more capable and complex over the years, the software associated with them has grown dramatically. For example, the F-4 aircraft of the Vietnam war era had practically no software. Today's F-14D aircraft currently relies on over one million source lines of code (SLOC) to perform its mission. In the near future,

estimates predict that the Advanced Tactical Fighter will require approximately seven million SLOC to operate.[Ref. 6] This growth represents an increase not only in volume, but also in software complexity. Complex software costs more to develop and support after fielding. Similar increases in software volume and complexity are evident in every category of system that depends upon Mission Critical Computer Resources (MCCR).[Ref. 1:p. 9]

The total amount of software demanded by DOD is staggering. A technical report by the Software Engineering Institute (SEI) estimated the DOD demand for Ada language alone in 1989 was over 40 million lines of code, requiring a rough estimate of over 9,000 person years of programming effort based on moderate code difficulty.[Ref. 7] This work estimated the number of lines of Ada programming code planned, in full scale development, and in the post deployment software support stage. Both figures are considered underestimates. When one considers the other MCCR application programs using languages other than Ada, the current amount of weapon system software is astonishing.[Ref.1:p. 9]

2. Software Costs

Producing this massive amount of weapon system software comes at no small cost to the Government. While cost data on DOD programs have been poorly tracked in the past, 1992 estimates of total software expenditures ranged from $24 billion to $32 billion. This amount was approximately 8-11% of the DOD budget for that year. In the next 15

years it is estimated that software may increase to an annual cost of $50 billion and account for up to 20% of the DOD budget.[Ref. 8]

The software developmental costs for software intensive systems can result in large portions of a weapon system program's budget.[Ref. 1] Table 1 provides some examples of the software developmental cost and its percentage of the total developmental cost of selected DOD MCCR systems [Ref. 8].

Table 1. Software Development Costs

SERVICE	PROGRAM	SOFTWARE DEVELOPMENT COST	% TOTAL DEVELOPMENT COST
Air Force	Adv Tactical Fighter	$1 Billion	13%
Air Force	B-1B Bomber	$726 Million	19%
Army	LHX Helicopter	$115 Million	3%
Navy	SSN-21 Submarine	$450 Million	13%
Navy	Trident II Missile	$280 Million	9%

With respect to volume, complexity, and cost, as well as functionality, software is a critical component in all of DOD's technologically advanced weapon systems. Software has grown into a multi-billion dollar facet of the defense procurement process and it clearly plays a critical role in DOD's quest to maintain technological superiority over U.S. adversaries.[Ref. 1].

C. SOFTWARE DEVELOPMENT PROBLEMS IN DOD

As software development has grown more complex throughout the years, so have the problems associated with its development. Software development problems have been referred to by some as a "software crisis."[Ref. 9] Air Force General Bernard Randolph has characterized software as the Achilles heel of weapon system development [Ref. 9]. The Defense Systems Management College's *Mission Critical Computer Resources Management Guide* describes the impact of software development problems on military weapon systems in this way:

> Most systems are delivered late, have cost overruns, rarely meet performance requirements upon initial delivery and are often ridiculously expensive to maintain. It would be unfair to blame all of these unpleasant facts just on digital systems and software, but it is generally recognized that software is a major contributor, and often the only contributor, to these problems.[Ref. 5]

A wide variety of software development problems plague DOD acquisition programs. There are many reasons why these software development problems have occurred and have persisted throughout the years.

Some of the more significant problems as outlined in various General Accounting Office (GAO) reports are listed below:

- Lack of management attention
- Inadequate requirements definition
- Requirements growth
- Integration deficiencies
- Inadequate assessment of contractors' software development and management capability

- Underestimation of software development risks
- Lack of adherence to software development standards
- Inadequate testing.[Ref. 10]

These problems contribute to significant schedule delays, cost increases, and performance shortfalls. The most disturbing fact about these problems is that they could have been avoided if proper emphasis had been placed on software risk management.

D. RISK AND RISK MANAGEMENT

Prior to discussing software risk management it is important to understand the concepts of risk and risk management. Webster defines risk as "the possibility of loss or injury." A risk is not a problem. To be technically precise, there are two factors that comprise a risk: probability or likelihood that it will occur and loss resulting from its occurrence.[Ref. 11:p. 3] The Defense Systems Management College (DSMC) guidebook, *Risk Management Concepts and Guidance*, provides an expanded definition of risk as the probability of an undesirable event occurring and the significance of the consequence of the occurrence [Ref. 12:p. 3-1]. With risk defined, it is also necessary to understand the term risk management.

Risk management can be thought of as an umbrella term for the processes used to manage risk [Ref. 12]. Typical processes of risk management are risk assessment and risk control with each of these processes involving subsidiary steps [Ref. 2]. The SEI defines risk management as follows:

> Risk management is really an ethic in which you (1) continuously assess what can go wrong, the likelihood of the event(s) happening, and the associated consequences should the event(s) occur; and (2) determine

alternative strategies to deal with the risks, study the impacts of those strategies, and choose which strategies to implement.[Ref. 13:p. 8]

Before addressing software risk management, some important DOD policy and guidance on risk management will be identified.

E. DOD GUIDANCE ON RISK MANAGEMENT

In the DOD acquisition process, risk management is required by policy. There are two major directives that provide guidance on risk management. Some of the major policy statements on risk management in these two directives is addressed next.

1. DOD Directive 5000.1

"Defense Acquisition," DOD Directive 5000.1, establishes a disciplined management approach for acquiring systems and materiel that satisfies the operational users needs. The directive addresses risk management by saying that risk management shall be a major consideration at each milestone beginning with the new start milestone decision [Ref. 14:p. 1-2]. Program risks and risk management plans shall be explicitly assessed at each milestone decision point prior to granting approval to proceed into the next acquisition phase [Ref. 14:p. 1-4].

2. DOD Instruction 5000.2

DOD Instruction 5000.2, "Defense Acquisition Management Policies and Procedures," implements the guidance provided in DOD Directive 5000.1. Part 5, Section B, "Risk Management," contains the policies and procedures which establish the basis for managing risk. This section directs that a risk management program shall be established for each acquisition program to identify and control performance, cost, and schedule risks.

The risk management program must include provisions for eliminating these risks or reducing them to acceptable levels. The instruction points out that industry participation in risk management is essential to ensure a clear understanding of program objectives, produce schedule realism, and identify appropriate incentives for contractual agreements. [Ref. 15:p. 5-B-1]

The instruction says the risk management program will consist of planning, identification, assessment, analysis, and reduction techniques to support sound program management decisions. Essential characteristics of the risk management program are that it will:

- Include a structured and documented risk assessment and analysis process, with user participation, to identify risks early in the program and to provide proactive, look ahead risk assessment and review.
- Include clearly defined criteria for elements leading to the risk assessment events.
- Include assessment of the contractor's managerial, development, and manufacturing capabilities and processes.
- Identify and track risk drivers, define risk abatement plans, and provide for continuous risk assessment throughout each acquisition phase to determine how risks have changed.
- Have clearly defined evaluation criteria for assigning risk ratings of high, moderate, or low to elements of risk associated with each major subsystem and the overall system.[Ref. 15]

Risks, risk reduction measures, and rationale and assumptions made in assigning risk ratings will be explicitly assessed at each milestone decision point as an integral part of this effort.[Ref. 15]

Concerning risk management with respect to computer resources, the instruction says the management approach, decisions, and plans associated with computer resources will be documented in the Computer Resources Life Cycle Management Plan (CRLCMP). The CRLCMP will identify all major computer resource risk areas, to include resources (people, training, facilities, funding, etc.), support risks, and software safety criticality and the methods for their control.[Ref. 15:p. 6-D-2]

Now that DOD guidance and policy on risk management have been identified, it is appropriate to look at the discipline of software risk management.

F. SOFTWARE RISK MANAGEMENT

Boehm [Ref. 2:p. 1] defines software risk management as a discipline whose objectives are to identify, address, and eliminate software risk items before they become either threats to successful software operation or major sources of software rework. Software risk management is important primarily because it helps people avoid disasters, avoid re-work, avoid overkill, and stimulate win-win situations on software projects.[Ref. 2:p. 1]

How and why did software risk management come about? The next section identifies how and why the discipline of software risk management evolved and what some of the initiatives are to improve the state of the discipline of software risk management.

G. THE EVOLUTION OF SOFTWARE RISK MANAGEMENT

Software risk management as a discipline is fairly new. A review of the literature indicates that prior to 1989, there were very few sources of information that described

software risks or how to deal with them. Capers Jones [Ref. 16] estimates that before 1990, failure to perform adequate risk assessments of software projects had been observed for 80% of all major (>1000 function point) projects observed. Since that time risk analysis has started to become more common under the combined impact of new books, journal articles, and new emphasis on risk management by groups such as the Software Engineering Institute (SEI).[Ref. 16:p. 254]

The new emphasis on software risk management seems to have been stimulated by the fact that the software field has had its share of disasters. People are looking for ways to avoid future problems. Boehm [Ref. 2] points this fact out by saying:

> The software field has had its share of disasters. Most post-mortems of these software disaster projects have indicated that their problems would have been avoided or strongly reduced if there had been an explicit early concern with identifying and resolving their high-risk elements. Frequently these projects were swept along by a tide of optimistic enthusiasm during their early phases, which caused project managers to miss some clear signals of high-risk issues that proved to be the project's downfall later.[Ref. 2]

1. Relationship of Software Risk Management to Risk Management

As mentioned in the previous section, the software field has had its share of failed projects. In the search for ways to avoid these disasters, people turned to the risk management field for answers. One of the first major works on the discipline of software risk management was written by Dr. Barry Boehm [Ref. 2]. In the preface to his book, *Software Risk Management*, Boehm relates the discipline of risk management to the newly evolving discipline of software risk management. Boehm says, "In the process of

researching software risk management as a discipline, I found that it can benefit from a long tradition of studying risk management in other situations"[Ref. 2: p. v].

Boehm then pointed out that the insurance business is founded on the ability to assess and deal with risk. Large corporations have risk management departments whose responsibility is to assess corporate risks and to establish appropriate risk management programs that involve various kinds of insurance, contract provisions, preventative measures, policies, and practices to deal cost-effectively with the corporation's risk exposure. He further points out that dealing with risk is central to the modern discipline of economics, particularly in such areas as decision theory, utility theory, and game theory.[Ref. 2:p. v]

Boehm's point was that material from these other disciplines provides software risk management with some valuable concepts and principles, but that its particular application to software and project management situations requires a good deal of tailoring [Ref. 2:p. vi]. At the time Boehm's work was published, he pointed out that there was no large body of software risk management literature to date. Given the critical leverage risk management can have on a project's success, Boehm thought it important to disseminate the information in his work as soon as possible.[Ref. 2:p. vi] The primary objectives of his volume were to enable readers to:

- Identify the major sources of risk on a given software project
- Understand the essential concepts and techniques involved in software risk assessment and risk control
- Apply these concepts and techniques to practical day-to-day software project situations.[Ref. 2:p. vi]

The effort of Boehm and other authors has added rigor to approaches to software risk management. In the last five years added impetus to software risk management has been provided by organizations such as the Software Engineering Institute (SEI).

2. Software Engineering Institute (SEI)

The SEI, located at Carnegie Mellon University in Pittsburgh, Pennsylvania, is a federally funded research and development center. Some of the SEI technical areas of focus are: software risk management, software process improvement, and software engineering techniques. The SEI has done extensive work in the area of software risk management on software intensive development programs in the last five years. An SEI objective in this area was to obtain knowledge in the area of software risk management and to publish the information periodically. A result of this effort was a series of technical reports addressing software risk management. An example of one of these reports is *Software Development Risk Management: an SEI Appraisal*.[Ref. 13]

The DOD directed the SEI to develop a means whereby the software process maturity of contractors could be evaluated [Ref. 17:p. 4.15-1]. The objective of this effort was to better attain management control of software development efforts, as well as improve production of high quality software. A key result of this effort by the SEI was the capability maturity model (CMM).

The CMM is a model whereby key process areas have been defined for different levels of process maturity. The objective of using the model is to assist a developer in achieving a desired level of process maturity while building a solid foundation of support at

each step along the way. The model is divided into five different levels as shown in Table 2 [Ref. 18:Fig. 1.2.1] below. A developer is assessed, evaluated, its strengths and weaknesses defined, and a program of process maturity improvement is established.[Ref. 17:p. 4.15-3]

Table 2. SEI Software Process Maturity Model

Level	Characteristic	Key Problem Areas	Result
5 Optimizing	Improvement fed back into process	Automation	Productivity & Quality
4 Managed	(quantitative) Measured process	Changing technology Problem analysis Problem prevention	
3 Defined	(qualitative) Process defined and institutionalized	Process measurement Process analysis Quantitative quality plans	
2 Repeatable	(intuitive) Process dependent on individuals	Training Technical practices - reviews, testing Process focus - standards, process groups	
1 Initial	(ad hoc/chaotic)	Project management Project planning Configuration management Software quality assurance	Risk

A major advantage which DOD derives from the application of the process maturity evaluation is the identification of program risks due to poor software quality products or immature development processes. The application of this process to the source selection process is called the Software Capability Evaluation process. This process offers the

Government an opportunity to minimize risk and uncertainty in software development when dealing with a particular contractor or development agency. This process can also be applied during the contract as a contract monitoring process.[Ref. 17:p. 4.15-3]

In addition to its work in the area of software capability evaluations, the SEI provides a variety of other services in the area of software risk management. Some of the related products and services are:

- Risk Identification and Analysis Course
- Software Risk Management Course
- SEI Conference on Software Risk
- Independent Risk Assessment (service)
- Software Risk Evaluation (service).

Toward the end of the thesis research process the author had the opportunity to participate in the execution phase of an SEI software risk evaluation (SRE). Appendix A provides the reader with some background information on the SRE process. It also describes the execution phase of the SRE process along with some of the author's observations.

The main purpose of this section has been to identify how and why software risk management emerged recently as a discipline. A secondary purpose was to show the relationship of software risk management to the discipline of risk management. The final item of interest addressed was the recent work the SEI has performed in the area of

software risk management. Next, the importance of software risk management to the DOD acquisition process will be described.

3. Software Risk Management and the DOD Acquisition Process

In the context of the DOD acquisition process, software risk management is extremely important. As discussed earlier, DOD policy mandates that risk management will be a part of every acquisition program. Software has become a multi-billion dollar endeavor for DOD, consuming about 10% of the Defense budget [Ref. 8]. The cost along with the added complexity of software make it imperative that acquisition managers have a process for averting software problems before they occur.

To many casual observers, it would seem that software risk management would be an important part of any software intensive weapon system acquisition. However, GAO reports over the past 20 years describe numerous weapon system programs experiencing years of delayed fieldings and substantial cost overruns. Potential software risk items were not effectively identified, addressed, and mitigated or eliminated. One recent case in point is the C-17 aircraft program. The situation was summed up succinctly in a GAO report:

> The Air Force made a number of major mistakes early in the program that affected its ability to manage and oversee software development. Air Force officials initially assumed that software development would be low-risk without performing the type of analysis necessary to support and document that assumption.[Ref. 19]

The program encountered significant software problems early in development and was unable to deliver the proper software for the initial test flight.

The essence of software risk management is to identify, address, and eliminate or mitigate software risk items before they become threats to successful software operation or major sources of software rework. To avoid software disasters, DOD acquisition managers must use the tools of risk management and software risk management. The key to success is to have a sound risk management program which includes software risk management. Having a sound risk management program requires a commitment from both the program office and the contractor(s). There is little documented evidence that sound software risk management practices have been integrated into the overall acquisition process for software intensive weapon systems. This thesis will focus on software risk management techniques being used by the V-22 acquisition program of the Department of the Navy.

H. SUMMARY

This chapter has provided background for the role that software plays in the dynamic environment of weapon system acquisition. The cost and complexity of software development dictate that special attention be paid to potential software risk items. Software risk management must be integrated into the acquisition management process for software intensive weapon systems as a way to identify, address, and eliminate or mitigate potential software risk items.

Next, Chapter III will introduce the V-22 weapon system and briefly detail the acquisition history of the program. It will also provide an overview of the major computer software configuration items (CSCIs). The transition from FSD to E&MD will be

discussed. Finally, Chapter III will address the important role of independent risk assessments in the program.

III. THE V-22 WEAPON SYSTEM

A. INTRODUCTION

The V-22 aircraft is being developed to perform various missions. These missions require the use of sophisticated flight control and avionics systems as well as other software intensive systems. The chapter will begin by describing the V-22 aircraft along with a brief history of its development. Next, the major computer software components of the V-22 will be identified and their functions will be described. The transition from the Full-Scale Development (FSD) phase to the Engineering and Manufacturing Development (E&MD) phase will be addressed. Finally, the important role of the independent risk assessment teams (IRATs) along with their findings and recommendations will be discussed.

B. THE V-22 WEAPON SYSTEM

1. Description

The V-22 will provide the Services with a multi-engine, dual-piloted, self-deployable, medium lift, vertical takeoff and landing aircraft to perform various missions [Ref. 20]. The V-22 is being developed to perform U.S. Marine Corps, U.S. Navy, and U.S. Special Operations Command combat missions for the year 2001 and beyond. The V-22 design incorporates advanced but mature technology proven in the XV-15 tiltrotor demonstrators, V-22 FSD models, and V-22 E&MD models. This technology includes composite materials, digital fly-by-wire flight controls, and advanced survivability and crashworthiness systems.[Ref. 21]

The V-22 will fill multi-service combat operational requirements including amphibious assault, land assault, medium cargo lift, combat search and rescue, Special Operations Forces support, and worldwide self-deployability. The aircraft will be capable of operations from aviation and air capable ships as well as from unimproved landing sites throughout the world. A tiltrotor combines the speed, range, and fuel efficiency normally associated with turboprop aircraft with the vertical take-off/landing and hover capabilities of helicopters. The tiltrotor aircraft represents a major technological breakthrough in aviation to meet both existing military needs, and through developmental growth, civilian applications.[Ref. 21]

2. Development History

In April 1986, the V-22 program passed Milestone II and entered Phase II, FSD. In May 1986, the Navy awarded a fixed-price-incentive-firm FSD contract with a ceiling price of $1,825 million to the team of Bell Helicopter Textron, Incorporated and Boeing Helicopter Company (Bell-Boeing) to design and produce six aircraft for flight and ground testing. Five of the six aircraft were produced but two crashed and were destroyed. The sixth aircraft was not fully assembled as a cost savings measure. The FSD contract also included an option to buy 12 aircraft under pilot production. Also in May 1986, the Navy awarded a firm-fixed-price contract with a ceiling price of $76 million to develop and produce engines for the FSD aircraft.[Ref. 22]

In April 1989, citing a lack of affordability, the Secretary of Defense deleted all funding after FY 1989 for the V-22 Program and requested funding for a mix of CH-53

and H-60 helicopters. However, Congress denied the Secretary's request and continued to fund the V-22 program. In June 1991, in response to a congressional mandate to obligate $200 million, the Navy awarded Bell-Boeing an FSD Phase II letter contract for $75 million, which was definitized in May 1992, as a cost-plus-fixed-fee contract. In July 1992, in an effort to resolve the continuing impasse between DOD and Congress, the Secretary of Defense proposed a solution to congressional leaders that involved developing and evaluating the V-22 and helicopters as alternatives to the medium-lift replacement requirement.[Ref. 22]

In October 1992, the Navy terminated the FSD contract and awarded a cost-reimbursable E&MD airframe letter contract to Bell-Boeing for $550 million ($558 million as of March 1994). In December 1992, the Navy awarded an E&MD engine letter contract to Allison for $65 million, which was definitized in September 1993, as a cost-plus-incentive-fee contract with a target price of $141 million. In May 1994, the Navy definitized the E&MD letter contract as a cost-plus-award-fee contract for $2.65 billion.[Ref. 22] The E&MD contract calls for the production of four aircraft. These aircraft will be used to continue the research, development, test, and evaluation program, provide aircraft for operational evaluation, and serve to demonstrate the production facilities for follow-on production contracts.[Ref. 21]

The V-22 Program is currently in the E&MD phase of acquisition. Aircraft number seven is scheduled to perform first flight in December 1996. Several phases of operational testing will be conducted using both FSD and E&MD aircraft over approximately the next

seven years. The testing will be conducted to assess the V-22's potential operational effectiveness and operational suitability. These tests will support a recommendation for fleet introduction and will also support the milestone III decision for the V-22.[Ref. 21:p. 121]

C. MAJOR COMPUTER SOFTWARE COMPONENTS

As with any new aircraft development, embedded computer software is an important consideration. The V-22 is no exception. It consists of several major computer software configuration items (CSCIs). The major CSCIs are the V-22 JVX Applications and Systems Software (JASS); V-22 JVX Simulation Support Software (JSSS); V-22 Flight Control Computer Operational Flight Program; Display Electronics Unit (DEU) software; Vibration, Structural Life, and Engine Diagnostics (VSLED) software; Interface Unit (IU) software; V-22 Maintenance Data Processing System (MDPS) software; and V-22 Mission Planning Station (VMPS) software [Ref. 21]. These CSCIs, along with some of the associated hardware, are described in detail in Appendix B. A summary of source lines of code, percent of code reused from FSD, and language for each major CSCI is presented in Table 3.

D. RESPONSIBILITY FOR SOFTWARE DEVELOPMENT AND INTEGRATION

The prime contractor team consists of Bell Helicopter Textron, Incorporated and Boeing Defense and Space Group, Helicopter Division (Bell-Boeing). Bell-Boeing is responsible for all aspects of V-22 development, including software development and integration.[Ref. 23]

Table 3. V-22 Computer Software Configuration Items. From Ref. [23]

CSCI	% NEW (Note 1)/ MOD (Note 2) SLOC	% REUSE (Note 3)	Language	SLOC
JASS Mission Computers AN/AYK-14(V) AMC	100%	0%	Ada	148K
Interface Units CV-4023/AYK, ABIU CV-4025/AYK, WIU CV-4026/AYK, NIU	 95% 95% 95%	 5% 5% 5%	80C186 C& Assembly C& Assembly C& Assembly	 6.2K (Note 4) 3.6K (Note 4) 4.0K (Note 4)
VSLED	15%	85%	PACE 1750 JOVIAL	17K
Flight Control Computers (Operational Flight Program)	22%	78%	PACE 1750 Assembly	173K
Display Electronics Unit (DEU)	80%	20%	68030 C& Assembly	100K
MDPS	New = 97% Mod = 1%	2%	Ada	111K
JSSS	New = 5% Mod = 50%	45%	Fortran	100K
TAMPS (VMPS)	New = 40% Mod = 40%	20%	C	50K (Note 4)

Note 1. NEW - % SLOC that were new development
Note 2. MOD - % SLOC modified from FSD for use in E&MD
Note 3. REUSE - % SLOC reused from FSD without any modification
Note 4. Denotes executable SLOC

The focus of this research is on the avionics and flight control system software. V-22 avionics system software is being developed at the Boeing Helicopters (BH) facility. The digital flight control system software is being developed by Boeing and Martin-Marietta Control Systems, the flight control system subcontractor.

E. TRANSITION FROM FSD TO E&MD

In October 1992, the Navy terminated the V-22 FSD contract [Ref. 22]. Toward the end of FSD and at the beginning of E&MD, two General Accounting Office (GAO) reports expressed concern about developmental problems that could make the transition to production a high risk [Refs. 24 and 25]. A 1990 GAO report referred to a production readiness review held in early 1989 by Naval Air Systems Command which identified concerns regarding a lack of software development that is essential for the proper functioning of the flight control system [Ref. 24:p. 2]. The report also cited cost growth attributable to several factors, one of which was the mission computer [Ref. 24:p. 5].

A 1994 GAO report also identified several development issues. Among these issues was a concern with incomplete development of software [Ref. 25:p. 16]. The DOD response concurred with the findings and stated that the purpose of E&MD was to correct the deficiencies noted during FSD [Ref. 25:p. 30].

The V-22 program is presently well into the E&MD acquisition phase, and the program office acknowledges that the continued development of the V-22 weapon system is a moderate risk program [Ref. 21:p. 99]. The primary challenges during E&MD from the perspective of the V-22 program office are: software development and integration,

weight reduction, affordability, producibility, configuration definition, and schedule concurrency. Software development and integration is categorized as a "moderate" risk by the program office.[Ref. 21:p. 99]

Software development is considered a moderate risk for several reasons. The development of the JASS software is a schedule risk for two primary reasons. First, the JASS rewrite effort is relatively large, and staffing in this critical area lagged behind schedule during the first year of the E&MD contract. Second, Bell-Boeing is converting from CMS-2 to the Ada programming language for JASS. The flight control system software development is also a schedule risk area. This assessment is based primarily on past performance in the development of software for aircraft two and three during FSD.[Ref. 26]

While the GAO reports identified some concerns regarding software development, more detailed insight on software development was gained from several IRATs which were chartered by the Program Executive Officer for Air Anti-submarine Warfare, Assault, and Special Mission Programs (PEO(A)) who has cognizance over the V-22 program.

F. THE INDEPENDENT RISK ASSESSMENT TEAMS (IRATs)

Within seven months of award of the E&MD letter contract to Bell-Boeing in October 1992, the PEO(A) chartered a V-22 Avionics System IRAT. Since the formation of the first IRAT in May 1993, three more IRATs have been chartered by the PEO. These IRATs served the important purpose of providing visibility of software development to the PEO in the V-22 program.

1. V-22 Avionics System IRAT of May-June 1993

The objectives of the IRAT were to identify avionics and software related risk areas, and to provide appropriate recommendations to the program manager [Ref. 21:p. 101]. The team consisted of approximately eight members distributed as follows: one from the PEO(A) staff, one from the Naval Air Systems Command staff, one from a Government field activity, and the rest from a contractor, Mitre Corporation. The team members had expertise in systems engineering and integration, software engineering, hardware technologies, and systems acquisition.[Ref. 27:p. 2] A summary of the IRAT findings is provided below.

- FSD system exhibited much of the required functionality, but the existing software was a poor foundation for E&MD.
- No show stoppers
    1. Avionics development was medium risk.
    2. Hardware and software were neither large nor complex.
    3. There was sufficient time in the schedule.
- Lack of critical skills and sound systems and software engineering processes were driving the risk high.
- Mitigation strategies existed to hold risk at medium.
- Fundamentals needed prompt attention.
- Indications of inadequate attention to software safety were sufficient to initiate an independent review.[Ref. 27:p. 4]

As a result of this IRAT, several recommendations were made for program actions, Government actions, and contractor actions that would, in the opinion of the team, reduce the risk in the avionics system development [Ref. 27:p. 5]. These recommendations are listed below.

- Program Actions

    1. Delay transition to Ada and the Advanced Mission Computer (AMC). After extensive study by the Government and the contractors, a decision was made in December 1993, to re-write the JASS software using Ada and to use an upgraded mission computer, the AMC. The IRAT thought that the contractor was unprepared to make an immediate transition; therefore they concluded that the lowest risk strategy would be to continue with the CMS-2 development using the FSD mission computer, the AN/AYK-14 VHISC Processor Module. The IRAT thought that the program could hold the risk in the Ada transition to medium if the contractor developed the necessary capability before proceeding. Therefore, the IRAT recommended that the program delay the transition to Ada.

    2. Charter an independent software safety review. Although the scope and depth of the review were insufficient to determine whether software safety problems existed in the Vehicle Management System (VMS), the IRAT observed deficiencies in the contractor's systems and software development processes that caused sufficient concern to warrant an immediate review.

- Government Actions. The IRAT recommended that the Government take several actions that they believed would reduce risk. The IRAT recommended that the Government reduce the potential for requirements creep and disagreements by developing a firm set of requirements and taking control of the allocated baseline. They also recommended that the Government define clearly the roles, responsibilities, and authority of the Government integrated product team (IPT) representatives and focus their participation in critical areas. The IRAT recommended that the Government add contractual requirements for normal development milestones (i.e., System Design Review, Software Specification Review) and associated documentation.

- Contractor actions. As a result of the 1993 review, the IRAT believed that it was critical that the contractors define, document, and implement a rigorous

systems engineering and software engineering process. On the basis of their review the IRAT concluded that the critical skills necessary to develop an adequate process were missing from the program, and they recommended that the contractors move quickly to add higher level systems and software engineering skills.[Ref. 27:p. 5]

Specific action taken on some of the above recommendations will be identified in Chapter IV where risk management on the V-22 program will be addressed in detail.

The Avionics IRAT recommended that there be an immediate, independent review of the VMS software and development process to investigate software safety. The IRAT thought a review was necessary because of the deficiencies they believed they had observed in the contractor's systems and software development processes. These deficiencies were sufficient to raise concern over the safety of the VMS software (in E&MD, the term VMS is a re-designation of the term FCS or flight control system).[Ref. 27:p. 4] As a result of this recommendation, the V-22 program office requested a further assessment of the Bell-Boeing (BB) flight control system (FCS) software development. Accordingly, an FCS Software Development Assessment Team was established in July 1993.[Ref. 28:p. 11]

2. V-22 Flight Control System Software Development Assessment Team of June-August 1993

This team was established with the specific charter to review the V-22 flight control system (FCS) software development and test process to assess its inherent effectiveness toward safety. The primary objective of the team was to look at the software development process for the FCS used during FSD, as well as the process proposed for E&MD. The team consisted of 10 members distributed as follows: one from NASA-Ames Research Center (Chairman), one from Naval Air Systems Command, four from Government field

activities, one from the Defense Plant Representative Office (Boeing site), and three from the software development contractors. The team members had expertise in software engineering, systems engineering, independent test and evaluation, flight control system engineering, software safety, and avionics system integration. The team's assessment focused specifically on process, not on existing software design.[Ref. 28] The major findings of the team are provided below.

- Systems engineering and software engineering were two FSD activities with serious shortcomings.

- A major concern was that the early FSD software development effort skipped over or gave light treatment to the planning and requirements phases and began directly with design and coding phase. The team pointed out that this approach leads to problems with traceability of requirements, maintainability of software, and potentially, system safety. The team further pointed out that requirements traceability is the cornerstone to successful programs, the heart and soul of regression testing, and the only reliable foundation for low-cost software maintenance. The mapping of requirements must propagate to the levels at which changes are made and tested.

- Lack of structured E&MD approaches to systems engineering, software engineering, and validation testing could not be remedied while the limited staff was engaged in actual testing, design activities, requirements definition, and other similar endeavors.

- Adequate time needed to be set aside to carefully plan the full range of E&MD software activities, to acquire experienced as well as new staff, to train the entire staff in the E&MD (as opposed to FSD) approach, and to incorporate the various support tools and tracking mechanisms to be employed during the E&MD software cycle. The team correctly pointed out that that this approach will save on cost and schedule in the long run. A change in software downstream is far more costly than properly specifying the software up front, and such a change adds an element of risk.

- During FSD, formalized analysis of the software for potential hazards was not required. The team pointed out that this is an industry wide problem because software safety analysis is a relatively new applied discipline. Software safety

analysis is particularly important to the V-22 aircraft because so many functions are controlled by software.[Ref. 28:p. 3]

The findings of the team led them to make several recommendations. The first recommendation identified action that could be taken to contractually bind the contractors to implement changes. The team recommended that the E&MD contract between the Government and Bell-Boeing (BB) include the following provisions.

- Permit no active design and/or specification of VMS software, targeted for E&MD aircraft, to proceed prior to Boeing Helicopter (BH) successfully:

    1. Establishing solid systems engineering, software development, and software testing guidelines, including a plan for a core regression testing approach.

    2. Implementing training of VMS staff in E&MD procedures and techniques.

    3. Implementing requirements tracing to the individual module and test case levels.

    4. Establishing strict configuration management for validation test cases, test results, and associated test tracing.

- Throughout the E&MD cycle, permit no authorization be made to the subcontractor (Martin-Marietta (MM)) regarding requirements or change implementation prior to BH updating all the pertinent requirement documents.

- Test facility upgrades

    1. Require that BB submit a plan and commit to a closed-loop, pilot-in-the-loop, flight control computer (FCC)-in-the-loop configuration, free of data or parameter transmission delays which would require that FCC software functionality be restricted. This configuration should also provide for an avionics/VMS interface test capability.

    2. Ensure adequate software validation test facilities are available to support both the V-22 risk reduction flight test activity and E&MD.

- Include all formal documents used by BH to transmit requirements to MM. Such documents need not require Government approval. Regular updates to these documents must be delivered to the Government.[Ref. 28]

In addition to the recommendations relating to contractual provisions, several other recommendations were made.

- BB should form a V-22 VMS software safety analysis team including appropriate Government and contractor representatives to: develop and update a software preliminary hazard list, prioritize the list, assist in determining potential hazard causes, and oversee the analysis and disposition of these hazards. Perform a software safety analysis of the latest V-22 FCC software being flown in the risk reduction aircraft. This action would be necessary in the interest of ensuring an effective software baseline through E&MD.

- BH improve VMS staffing through the following measures:

    1. Establish a small dedicated group whose primary responsibility is support of risk reduction activities.

    2. Increase the level of staffing with individuals experienced in modern systems and software development disciplines.

    3. Increase staffing in E&MD through establishment of two-person (as a minimum), dedicated teams in each of the following areas: software requirements, control laws, built-in test, redundancy management, executive/data management, test management, and flight simulation lab/ flight control system integration rig/avionics interface.

- Government and BB permanently establish a review team to monitor, on a regular basis, the E&MD software development effort. Team members should include representatives from each contractor, the Navy, and perhaps one or more outside groups.

- BH implement additional measures to improve communication with MM, to include a full-time, on-site, technical presence at MM.[Ref. 28]

It is important to point out that the team found no reason to declare the existing VMS software unsafe, but only that the process used to produce the software during FSD

might have exposed the product to risk [Ref. 28:p. 4]. The team also noted that improvements had been initiated in several areas and should continue [Ref. 28:p. 4]. Specific actions taken by the program office and the contractor with respect to this assessment will be identified and discussed in Chapter IV where program risk management is addressed.

In an effort to review progress made since the 1993 Avionics System IRAT, the PEO chartered another Avionics System IRAT in 1994.

3. V-22 Avionics System IRAT of August 1994

The objectives of the IRAT were to review the avionics development plans and processes, and to assess the progress made since the 1993 IRAT review. With one exception, all the team members of this IRAT participated in the Avionics System IRAT of 1993. The main goal of the assessment was to assist the program management team in planning and managing the program to ensure success.[Ref. 27:p. 3] The team took the approach of highlighting potential problems and offering general strategies for risk mitigation and improvement [Ref. 27]. A summary of the findings of the IRAT is provided below.

- Management had taken positive action.
- Substantial improvements had been made.
    1. Processes and structure were maturing.
    2. Staffing in critical skill areas improved.
- Development remained manageable.

1. Task scope had not increased.

2. Adequate schedule remained.

- Continued attention to important areas was needed to maintain the positive trend.

1. Monitor avionics software and advanced mission computer development closely.

2. Continue systems and software process improvements.

3. Continue software safety process development.[Ref. 27:p. 15]

The IRAT pointed out that during the 1993 assessment they felt that the avionics development represented a medium risk that would become high very quickly without corrective action. Because of the positive action taken they believed that the trend had been reversed and the situation was improving.[Ref. 27:p. 15] This trend was due in large part to the V-22 E&MD risk management process which will be discussed in detail in Chapter IV.

One of the recommendations from this assessment was that the V-22 program should address the issue of a software safety review of the FSD flight control system software product. The team pointed out that both the 1993 Avionics System IRAT and the V-22 Flight Control System Software Development Assessment Team recommended a review of the product.[Ref. 27:p. 18] Accordingly, a V-22 Digital Flight Control System (DFCS) Software Product Assessment Team (SPAT) was chartered in October 1994.

4. V-22 FSD Digital Flight Control System (DFCS) Software Product Assessment Team (SPAT) of October-December 1994

The scope of this product assessment was limited to the FSD DFCS software and its related artifacts. The goal of the assessment was to be able to determine what the risk level was to the V-22 program relative to the FSD DFCS software and to determine any necessary corrective actions relative to the DFCS software that the V-22 program should take to reduce its risk.[Ref. 29:p. 3] The team consisted of nine members distributed as follows: five from Government field activities, one from Naval Air Systems Command, two from software development contractors, and one from an independent contractor. The team members had expertise in software engineering, software safety, and flight control system engineering.[Ref. 29]

The SPAT did not validate critical functions of the FSD DFCS software, rather it evaluated the software's structural integrity and the process compliance of the FSD system within which safety critical software had been developed [Ref. 29:p. 4].

Although there were many positive areas observed by the SPAT, there were several areas of concern. Three areas of concern were: testing, software product inconsistencies, and software configuration management/software quality assurance.

- Testing

    1. No evidence of stress testing. The SPAT defined stress testing as "testing to exercise the error handling and reaction to unanticipated inputs to the software at all levels." Testing was oriented more towards verification and validation, vice stress testing to determine if unusual conditions could cause the software to fail.

    2. No evidence of regression analysis. Regression testing at MM appeared to consist primarily of running the unit test for each module that had

been changed. For modules that had an integration test, the integration test was also run for the module that was changed. There was no evidence that a regression analysis was performed to determine if this level of testing was sufficient, or if other modules may have been affected indirectly by the change.

3. No evidence of requirements traceability. Requirements were not traced through to the actual software code. In looking at the source code file of a particular module, one could not find the requirement (at any level) for the function performed by that module. As a result, testing for compliance to requirements is a tedious and difficult task.

4. Limited software integration testing. Software testing using the 1750A simulator at MM was performed at the unit level only. No integration (regression) testing of groups of modules was performed using the simulator.

5. Evidence of non-compliance in unit testing. The unit tests for four modules were examined in detail. For all four modules the tests were found to be weak and did not satisfy the current documented MM unit test guidelines. There was no evidence of stress testing.

6. No documented hardware/software integration level test selection criteria. The process used in FSD for performing regression testing was not documented; there was no list of hardware/software integration tests or system level tests that need to be run when a module is changed (there is a unit test associated with each module) to ascertain whether interfacing modules still operate correctly.

- Software product inconsistencies

  1. Coding conventions. The SPAT identified certain violations of coding conventions and inconsistencies in internal code documentation. The SPAT found a large number of potential inconsistencies in the use of freeze/release comments. The documentation (code comments) of what registers are modified in a module were sometimes incomplete, especially when the transitive closure of all called macros and modules are considered. In the source code and the design documentation that the SPAT reviewed, the conventions for logical values were not documented.

  2. Hierarchical Input Process Output (HIPO) versus code. Seven modules were examined to ensure that all design requirements in the HIPO were

implemented in the code. For two of these modules discrepancies were found between the HIPO and the code.

3. Internal inconsistencies in design documentation and in code. In addition to inconsistencies between the HIPO and the code, inconsistencies within a code module or within a HIPO were found. Numerous cases of erroneous comments were found within the code.

4. Test reports. In trying to retrieve three hardware/software test reports one of the reports could not be found.

- Software configuration management/software quality assurance. There was a lack of rigor in some code and unit test review activities. It was found that peer review for code and unit test did not require the use of formal checklists. In fact, formal checklists are not required unless more than 50% of the unit software is changed.[Ref. 29]

The recommendations that resulted from the assessment are highly technical in nature and beyond the scope of this research. However, it is useful to point out that the SPAT served the important purpose of identifying areas of concern with the FSD DFCS software. The SPAT also made recommendations to address concern over potential risk areas with the goal of keeping the DFCS software at the lowest possible risk in terms of safety [Ref. 29:p. 34]. It is also important to point out that the SPAT found no deficiencies that would indicate unsafe operation [Ref. 29:p. 34].

This section has discussed the important role of independent assessments in identifying software related risk areas. From early 1993 through the end of 1994, these assessments have provided the cognizant PEO with extremely important insight as to the status of software development. The assessments have also provided insight on potential problem areas where action was needed.

G. SUMMARY

This chapter has provided a description of the V-22 weapon system, a system that will fill multi-service combat operational requirements. A brief history of the development of the V-22 was detailed. Major software components were identified and their functions were described. The transition from FSD to E&MD and some concerns relating to software development were described. Finally, the important role of the IRATs along with the results of their assessments was provided.

The next chapter will describe the V-22 risk management process. It will also describe how software risk management is implemented as part of the overall risk management process. Key Government and contractor actions which have affected software risk management will be identified and discussed. Also, an example of risk management will be provided.

IV. RISK MANAGEMENT IN THE V-22 PROGRAM

A. INTRODUCTION

In order for the risk management process to work, it must become formal, systematic, and be applied in a disciplined manner [Ref. 12:p. 2-1]. There are no "cookbook" solutions to risk management [Ref. 30]. Each situation is different and each circumstance requires a slightly different approach [Ref. 12]. As stated in Chapter II, the essence of software risk management and risk management in general is to identify, address, and eliminate or mitigate risk items before they become threats to program success. One of the major keys to success in program management is having a sound risk management program which includes software risk management. Having a sound risk management program requires a commitment from both the program office and the contractor(s).

This chapter begins by examining the risk management philosophy and risk management process in the V-22 program. Next, the implementation of software risk management will be described along with some of the factors that have significantly impacted the software risk management process. Key Government and contractor actions taken with respect to software risk management will be identified and discussed. Finally, an example of an identified risk item will be provided along with the approach that is being used to track and mitigate the risk involved.

B. RISK MANAGEMENT PHILOSOPHY AND PROCESS

The V-22 Program Manager (PM) considers sound risk management to be imperative to the success of the program [Ref. 26:p. D-3]. This section will examine the foundation for the V-22 risk management process as well as the process itself.

1. The V-22 Risk Management Philosophy

V-22 program management sold the E&MD phase of the program as a phase that would produce production representative aircraft. The PM realized that a significant amount of funding ($4 billion through fiscal year 1993) had already been devoted to development in the FSD phase of the program [Ref. 25]. A significant amount of funding would also be devoted to tooling and other investment for production during the E&MD phase of the program. This realization was a prime motivating factor for an aggressive risk management program.[Ref. 31] It is in this environment that the V-22 risk management philosophy was developed.

The V-22 risk management philosophy is to create an open, honest, risk-aware culture in which risk management is considered to be a normal, healthy aspect of overall program management [Ref. 32:p. 1]. This management philosophy has aided tremendously in ensuring that an environment exists where risks are freely communicated throughout the program management structure. For risk management to work, more than just a philosophy is required. As stated in the introduction to this chapter, the risk management process must become formal, systematic, and be applied in a disciplined manner to be

successful [Ref. 12]. The V-22 program has developed just such a process for risk management.

2. The V-22 Risk Management Process

The need to meet technical performance requirements and perform within program cost and schedule constraints dictates that risk be managed in a controlled, systematic manner. As the V-22 program entered E&MD, the Deputy Program Manager for Production (PMA-275D) established a much more formal process for risk management than the one that was in place during FSD. PMA-275D worked with the contractors to establish a formal, documented process.[Ref. 31] The risk management process has evolved substantially during E&MD and is documented as a formal program procedure [Ref. 32]. The formal document provides direction to all program and supporting organizations regarding the risk management process for the V-22 Osprey program. The procedure has been fully coordinated between the customer (V-22 program office) and the contractors, Bell-Boeing (BB) [Ref. 33].

PMA-275D is the V-22 program office focal point for risk management and heads the V-22 risk management team [Ref. 31]. Team membership is drawn from various integrated product teams (IPTs) within the Naval Air Systems Command matrix organization [Ref. 23]. Support contractor personnel, who assist the V-22 program office in the area of risk management, are also on the team [Ref. 30]. The team consists of a core of approximately 10 members and meets on a weekly basis to update status on outstanding risks. Attendance at the weekly meetings varies based on the number and nature of

outstanding risks.[Ref. 30] Including the time spent by PMA-275D, the author estimates that the team spends approximately 250-300 hours per month on risk management.

The V-22 risk management process consists of risk identification, risk characterization/analysis, impact assessment, development of risk reduction plans/mitigation strategies, implementation of action plans, monitoring progress, and disposition of risk items [Ref. 34]. Each element of the process is briefly discussed below.

- Risk Identification. Risks are primarily identified through the integrated product team (IPT) process, but can also be identified by customer or contractor program management. Risks can also be identified through various other means such as: schedule network analysis, test results, meetings/ discussions, review of IPT minutes, technical compliance matrix, technical performance measurements, and review of cost, schedule control system data. [Refs. 32 and 34]

- Risk Characterization/Analysis. As potential risks are identified, the items should be detailed on a potential risk form, which is included as an attachment to the program procedure. The potential risk is then forwarded to the Bell-Boeing (BB) program office and subsequently distributed to the customer and the Risk Management Control Board (RCB) focal points at each site. The potential risk is then reviewed at the next RCB meeting and a decision is made as to whether or not the item should be formally tracked as a risk. The review includes both a qualitative and quantitative assessment. If it is decided that the item should be tracked as a risk, it is assigned a permanent identification number, a level, and a category and added to the risk database.[Ref. 32] The RCB and the risk database will be addressed in more detail later in this section.

- Impact Assessment. Alternatives for reducing the risk are identified and analyzed. A three-fold impact assessment is conducted: impact if no action is taken; impact based upon most likely course of action; and impact of the action itself. The impact is documented in the E&MD risk database.[Ref. 34]

- Development of Risk Reduction Plans/Mitigation Strategies. After a risk has been identified and an identification number assigned by the RCB, a risk abatement plan must be developed by the responsible IPT. The amount of detail that is required for a risk abatement plan is dependent upon the level of the risk assessment. All risk plans are developed, reported, and monitored

using a common format which is provided as an attachment in the program procedure. Once the abatement plan has been developed, it is presented at the RCB for final review and approval. Any issues with the plan are resolved at this time by the RCB. Disapproved plans are either closed or revised based on RCB inputs.[Ref. 32] An example of an actual risk abatement plan is shown in Appendix C.

- Implementation of Action Plans. Risk abatement plans are authorized for implementation with RCB approval. The RCB obtains concurrence, if required, from the V-22 PM, Bell-Boeing PM, or procuring contracting officer. At the time of plan approval, the risk is assigned a permanent identification number and the date when the next plan update is required is established. This date is based upon the planned completion of events in the abatement plan. As a minimum, status will be provided on the moderate and high plans on a monthly basis. During this review, management actions will be assigned to address any issues that must be resolved in order to execute the plan (i.e., provide funding for an alternate plan, resolve a resource issue, etc.).[Ref. 32]

- Monitoring Progress. All risks are monitored through the IPT process and are reviewed by the RCB. Additionally, risk status is examined in design reviews, presented monthly to program management, and reviewed in program management reviews.[Ref. 32]

- Disposition of Risk Items. Risk items are assessed weekly by the V-22 risk management team. Items are downgraded or upgraded as appropriate. Risk items are also categorized as open, closed, or monitor. Closed items are retained in the risk database.[Ref. 34]

Two important components of the risk management process are the RCB and the risk database.

The RCB is an integrated management team consisting of representatives from the customer and BB. It is responsible for overseeing the risk management process for the program. The board meets every two weeks and consists of a core group of members with others attending as required based on subject matter. The core group consists of representatives from the customer and BB. The V-22 PM and the BB Project Officer (PO)

serve as the board co-chair. A BB representative records the minutes and documents all decisions made during the meeting. These minutes require Naval Air Systems Command and BB approval and are distributed within five working days of the meeting. The meetings are held via face-to-face sessions, video conferences, or teleconferences as the subject matter warrants.[Ref. 32]

A common risk database is acknowledged by both the customer and BB as the common source of detailed risk data for the program. This database is maintained by the Government through a coordinated effort with BB, the customer, associate contractors, and other Government agencies.[Ref. 32] An excerpt from the V-22 E&MD risk database is shown in Table 4.

As part of the formally documented process, roles and responsibilities for risk management are clearly defined. The roles and responsibilities are defined below.

- IPTs

    1. Identify and categorize risks.
    2. Prepare abbreviated risk plans.
    3. Prepare risk abatement plans for high and moderate risks.
    4. Identify resource/funding requirements for risk abatement.
    5. Implement and provide status on risk abatement plans following approval by program management.
    6. Review risk status as part of the weekly IPT agenda and provide status to the analysis and integration teams (AITs).[Ref. 32]

ID	Desc	Type	Reason	Status	Impact	Action	Level
41A	FCS Software Development	S	Past performance illustrates this is a risk area.	August sotplight charts show all areas satisfactory or excellent. Version 2.2 coupled AFCS validation completed (07/31/95) to support an October flight. First E&MD software version 10.0 is installed in FCSIR. FCSIR testing started.	Delay in aircraft #7 first flight date and FCSIR testing.	- Monitor VMS software metrics - Sep stoplight chart. Next action due date 09/28/95.	Mod
105	AMC	T	Advanced Mission Computer with LAMPS ROSP is new to V-22.	A detailed plan has been developed that supports V-22 needs (independent of LAMPS). In addition, Boeing has sent out requests for information to other vendors as a potential backup plan to develop alternate mission computer. Last AMC block 1 delivery has slid to (09/19/95). ROSP full 1553 drop has slid to (10/23/95). SOF qualification TRR has been completed with testing scheduled to start (10/31/95). LBC-02, MCC-02 ASIC prototype delivered (08/29/95). Current risk reduction schedule shows ROSP full 1553 drop scheduled (09/14/95), program status review of (08/18/95) indicated this could slip to (10/23/95). Last block 1 delivery scheduled (09/19/95) and ROSP block 1 integration to be completed (09/14/95).	Delay in Aircraft #7 first flight.	- Monitor progress. - Last AMC block 1 delivery. Next Action Due Date 09/19/95.	High

Type: T = Technical
S = Schedule
C = Cost
Open Plan Date as of: 09/07/95

Table 4. Excerpt From V-22 Risk Management Database

- AITs

    1. Task the IPTs with identifying risks and developing risk abatement plans for risks which are generated internally and risks which are flowed from the customer through the RCB.

    2. Review all risk abatement plans submitted by the IPTs.

    3. Review any additional IPT resource/funding requirements for risk abatement, identify the source, and provide approval as required.

    4. Review risk status with program management and receive approval from program management for resource/funding, as required.

    5. Provide risk plans and status to the PO risk focal point.[Ref. 32]

- BB Program Office

    1. Serve as the Bell-Boeing focal point for risk management and co-chair of the RCB.

    2. Develop and maintain a procedure for the risk management process.

    3. Maintain the V-22 risk list for the program.

    4. Keep the customer abreast of changes to the risk abatement plans.

    5. Develop agenda and distribute materials for the RCB meeting.

    6. Publish RCB meeting minutes within five working days of the meeting.[Ref. 32]

- Risk Management Control Board

    1. Provide direction and guidance for the risk management process for the V-22 E&MD program.

    2. Develop appropriate metrics for tracking and reporting the status of risk activity.[Ref. 32]

The IPTs and AITs play an important role in the risk management process for the V-22 program. Appendix D further describes the structure and roles of the AITs and IPTs.

3. Program Manager (PM) Use of Risk Management Information

The V-22 PM relies heavily on the risk manager, PMA-275D, and the RCB to manage V-22 program risks. The PM receives risk information on an exception basis.[Ref. 35] The V-22 risk manager and RCB mutually determine which risks are identified to the PM. Status is provided to the PM on high and moderate risks on a monthly basis. Also, the V-22 risk manager provides weekly updates on high risk items. The PM uses this risk information to make management decisions such as re-allocation of resources or schedule adjustments.[Ref. 35]

The risk management process has helped eliminate monthly and quarterly major program reviews. At major program reviews, which occur every six months, top risk items are discussed. Also, the PEO is briefed quarterly by the PM and top program risks are communicated at this briefing.[Ref. 35]

This section has examined the V-22 risk management process. The program risk management philosophy was described as a foundation for the structured, documented risk management process. The actual risk management process was also discussed. Finally, PM use of risk management information was discussed. The V-22 risk management process can be viewed as a fully integrated process with extensive contractor involvement and clearly defined roles and responsibilities.

As mentioned in the introduction to this chapter, software risk management is considered as part of the overall risk management program. The next section explains how software risk management is implemented within the V-22 program structure. It also describes some of the factors that have affected the software risk management process.

C. SOFTWARE RISK MANAGEMENT IN THE V-22 PROGRAM

In the V-22 program the software risk management process is fully integrated into the overall risk management process. This section will explain how the software risk management process is implemented within the V-22 program structure using the AIT and IPT.

There have been several factors which have affected software risk management in the V-22 program. These factors are listed below and will also be discussed in this section.

- Use of software metrics as a risk management tool
- Management attention
- Personnel experience/competence
- Contract type

1. Implementation of Software Risk Management

Appendix D explains in general terms what the AITs and IPTs are and how they operate. The V-22 risk management process flows responsibility for risk identification, tracking, and mitigation to the AITs and IPTs.

One of the key AITs in the software risk management process is the Avionics AIT. The V-22 procedure on risk management is implemented for the JASS software CSCI in the Avionics AIT by:

- Identifying and tracking risk items with risk reduction profiles. An example of a risk reduction profile is provided in Appendix E.

- Flowing risk management requirements to the IPTs and subcontractors.

- Elevating appropriate risks to the Air Vehicle AIT.[Ref. 36]

As depicted in Figure 2, the Operational Software IPT is one of seven IPTs under the Avionics AIT. The Avionics Operational Software IPT has implemented risk management by:

- Documenting risk management in section 3.3 of the Software Development Plan.

- Identifying Avionics Operational Software IPT action items in weekly meetings with the Avionics AIT.

- Elevating appropriate risk items to the Avionics AIT.[Ref. 36]

The IPT/AIT process for software risk management on the other CSCIs in the V-22 weapon system is similar to that of the JASS CSCI described above. A benefit of IPTs in software risk management is that there is Government membership and participation in each IPT. Having a Government representative on each IPT facilitates the early identification of risks. Government representatives participate in all IPT meetings. Therefore, when risks are surfaced in IPT meetings the Government has immediate

visibility. Early visibility allows action to be taken before a risk becomes a problem.[Refs. 30 and 37]

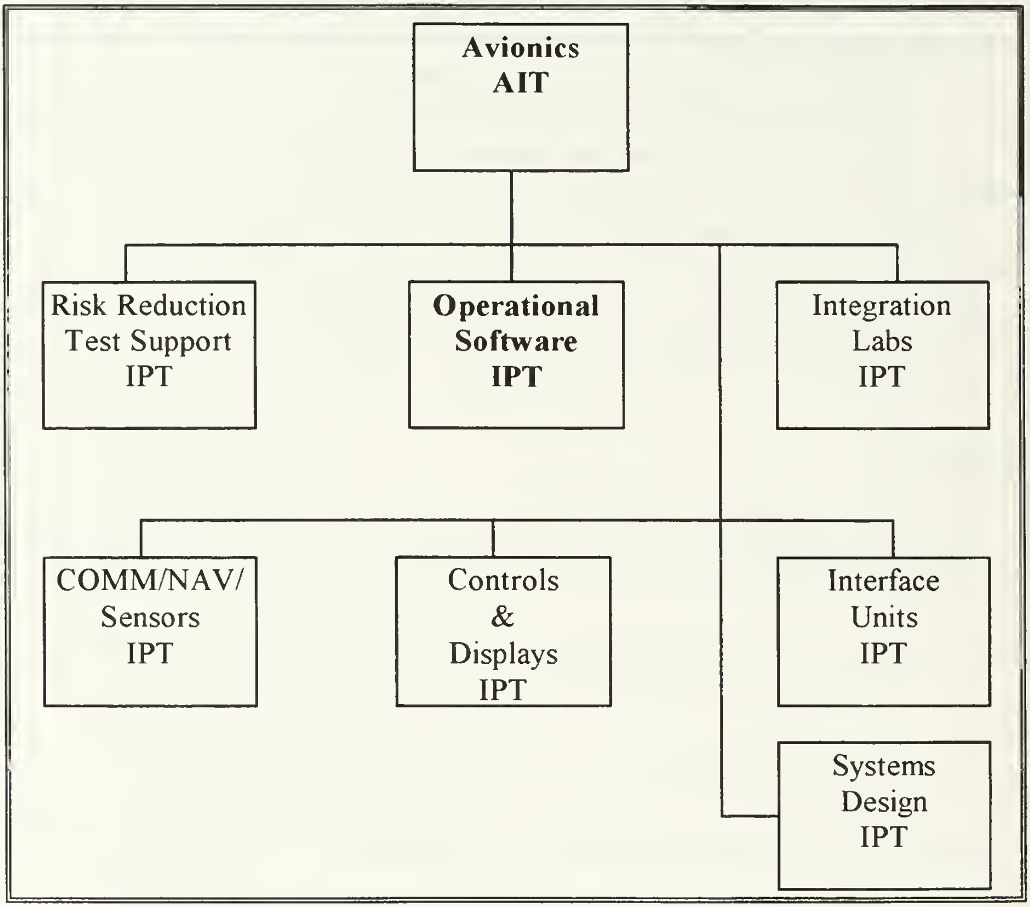


Figure 2. Avionics Integrated Product Teams. From [Ref. 38]

2. Use of Software Metrics as a Risk Management Tool

At the beginning of the E&MD phase, software metrics were recognized as being essential management indicators necessary to track software development. In the V-22 program a variety of software metrics are submitted by the contractors to the program office. These metrics are analyzed by program office engineering personnel to determine

the status of software development. The metrics are used not only to determine past and current performance but are also used to predict future performance. The metrics, when properly used, can be good predictors of future problems, thus making software metrics an important risk identification tool.

Metrics are contractually required and are submitted by the contractor on a quarterly basis in the Software Development Status Report (SDSR). The SDSR presents a series of graphs and charts that provide a top level summary of the software development effort. Some of the metrics provided are: memory resource utilization, processing time utilization, software designed, software coded, software units tested, and closed problem/ change reports.[Ref. 23] This list is not all inclusive but is representative of the type of metrics collected.

During the E&MD phase, the need for additional metrics was identified. As a result, metrics were developed using a prime contractor engineering operating instruction (EOI). The EOI provides instructions for a consistent and uniform approach for the collection of metrics. The framework for software status is the avionics stop light chart shown in Figure 3. The chart serves as a composite graphic for metric indicators. The stop light chart is supported by lower level metrics. For example, the product factor is a cumulative rating that is substantiated by the following indicators: software size, requirements growth, requirements definition and stability, and incremental release content. In addition, each factor on the chart is weighted.[Ref. 23]

	Weight	Dec 94	Jan 95	Feb 95	Mar 95	Apr 95	May 95	Comments
Cost	15%	2	2	2	2	2	2	
Schedule	15%	1 ↑	1	1	1	1	0	Mission Computer HW & SW schedules and JASS development
Product	10%	2	2	2	2	2	2	
Process	10%	2	2	2	2	2	2	
Tools	10%	2	1	1	1	0	1	Computer Resources @ BH are being installed
Personnel	20%	2	2	2	2	2	2	
Equipment	10%	2	2	2	2	2	2	
Training	5%	3	3	3	3	3	3	
IPT Process	5%	3	3	3	3	3	3	
Composite Rating		1.95	1.85	1.85	1.85	1.75	1.70	

Composite Range	0 Critical (0-0.9)	1 Unsatisfactory (0.9-1.7)	2 Satisfactory (1.7-2.4)	3 Excellent (2.4-3.0)	Forecast: ↓ Getting Worse ↑ Getting Better

Figure 3. Avionics Stop Light Chart. From Ref. [23]

The stop light charts are not contractually required but are delivered on a monthly basis for evaluation.[Ref. 23] These charts provide a summary of past performance as well as a forecast of future performance. Assessment of metric indicators is a primary tool for software risk management in the V-22 program.[Ref. 39]

3. Management Attention

From the early stages of the E&MD phase, management attention has clearly been focused on software development as a risk to the V-22 program. Upper level management attention came mainly from the PEO and PM level.[Refs. 27, 28, 29, and 31]

Due to program performance in FSD [Refs. 24 and 25], the PEO with cognizance over the V-22 program felt that it was necessary to gain detailed insight on the outlook for software development early in the E&MD phase. As a result the IRATs discussed in Chapter III were conducted. The IRATs were chartered to identify avionics and flight control system software related risk areas, and to provide appropriate recommendations to the V-22 PM. These assessments highlighted potential problem areas relative to software development and integration, systems engineering, software engineering, documentation, and manpower. Key actions taken by both the Government and contractors with respect to the software risks identified will be addressed later in this section.

In addition to management attention at the PEO level, attention to software development has also been focused at the PM level. In the V-22 program, the PM has realized that software development is typically not well understood by program managers. Therefore, the management philosophy is that software should always be monitored closely

and considered at least a moderate risk.[Ref. 31] A by-product of the fact that software is not well understood at the program manager level is the fact that someone must be found who is "smart" (technically competent) in software to manage software development for the program office [Ref. 31].

4. Personnel Experience/Competence

A necessary ingredient to the success of the V-22 program or any other software intensive program is to find someone who is competent in the area of software to manage the aspects of development related to software.[Ref. 31] The V-22 program has an experienced and highly competent Government Avionics Systems Project Engineer (ASPE) who is responsible for development of avionics software and hardware for the program.[Ref. 31]

The ASPE has 15 years of experience in the civilian sector in jobs ranging from electronics technician up to full engineer status. The ASPE received an undergraduate degree in computer science with a minor in electrical engineering while working in the civilian sector. He also has software experience in the civilian sector in the area of simulation using assembly language and C. Prior to coming to the V-22 program the ASPE had management level experience with the Government on other avionics development programs. The ASPE's Government education in software is limited to a systems engineering course completed at DSMC. While a member of the Computer Resources Division at Naval Air Systems Command, the ASPE attended monthly training sessions in software related topics such as: managing software changes, metrics, software

testing, configuration management, and independent validation and verification. The ASPE now considers himself a hybrid with formal education and work experience in both hardware and software.[Ref. 40]

Program management expressed a high level of confidence in the ASPE to effectively manage the risks of software development. In addition to confidence in Government personnel, program management also has a high level of confidence in the contractor software managers in the areas of avionics and the flight control system. In the E&MD phase the contractors have taken an aggressive approach to identifying and addressing software development risks. These managers have developed an effective strategy, including the use of metrics, for dealing with the risks of software development.[Ref. 31]

5. Contract Type

The type of contract used during FSD was a fixed-price type contract [Ref. 22]. Due to the fixed-price environment of the FSD effort there was little opportunity for the Government to significantly influence or oversee the software development process. This environment may have been good for expediting first flight, but resulted in shortfalls in several areas related to software development such as systems engineering and validation testing.[Ref. 28]

In May 1994, the E&MD contract was definitized as a cost-plus-award-fee (CPAF) contract [Ref. 22]. This type of contract gives the V-22 program the flexibility to provide incentives in certain areas where additional emphasis is desired from the contractor.

Because the V-22 Program Manager considers risk management to be imperative to the success of the program, the V-22 team has designated risk management as a major award fee criterion during the E&MD effort.[Ref. 26] This approach gives the contractor an incentive to demonstrate an active risk management program that identifies areas of risk and provides for corrective action.

D. KEY GOVERNMENT ACTIONS

As the V-22 program entered the E&MD phase, the PEO with cognizance over the program chartered several IRATs to identify software related risk areas and provide appropriate recommendations to the PM. These recommendations were identified in Chapter III. As a result of these recommendations the Government took several actions that would reduce the risk in software development during the E&MD phase. This section will discuss some of the key actions taken by the Government in response to those recommendations.

1. Transition to Ada and the Advanced Mission Computer (AMC)

In FSD, the JASS software CSCI was developed using the CMS-2 computer language. After extensive research and study by the Government and the contractors, a decision was made in December 1993, to re-write JASS using Ada [Ref. 21]. The JASS CSCI will be embedded in the AN/AYK-14(V) AMC [Ref. 21]. The AMC is a new item being developed as a spin-off from the Light Airborne Multi-Purpose System (LAMPS) integrated mission processor [Ref. 23].

The change that concerned the 1993 Avionics IRAT the most was the proposed switch to Ada and the AMC for JASS. The IRAT thought that the contractor was not prepared to execute an Ada development because of a lack of staff experienced in Ada software development and the immaturity of their software development process. The IRAT recommended that the program delay the transition to Ada and continue with the CMS-2 development to reduce the risk in meeting the first flight milestone.[Ref. 27]

Because of the cost implications and other difficulties in maintaining parallel CMS-2 and Ada developments, the program elected to accept the increased risk and proceeded directly to the Ada system [Ref. 27].

The V-22 program took aggressive action to reduce the risk associated with the transition to Ada/AMC. A V-22 risk mitigation strategy for the transition from CMS-2 to Ada has been developed and followed.[Ref. 27] The risk associated with JASS development is being reduced through close monitoring of carefully selected software metrics, including manpower considerations and Government participation in the software IPTs. Also, a number of periodic in-process reviews have been scheduled to ensure that progress in this critical area is satisfactory.[Ref. 26] Contractor action will be discussed under key contractor actions.

2. Requirements

The 1993 IRAT recommended that the V-22 program develop a firm set of requirements and take control of the allocated baseline [Ref. 27]. Four major builds of the avionics software are planned. The specific requirements for each software build were

defined and frozen at the critical design review. Any changes which may occur now that CDR is complete are subject to a rigorous review and approval process.[Ref. 27]

3. Clarification of Government Roles and Responsibilities

The 1993 Avionics IRAT review was conducted at a point early in the transition to an IPT management structure. The IRAT found that the management process was very immature and disorganized.[Ref. 27]

The management process has matured significantly and continues to evolve. The program has formed new IPTs and is also augmenting the IPT structure by forming working groups and tiger teams to address cross-cutting issues. The program management team has clarified and communicated the role of the Government IPT representatives. Also, the Government has better integrated its field support activities with the IPTs. As a result of this action there is now very little confusion over the Government role with respect to IPTs.[Ref. 27]

E. KEY CONTRACTOR ACTIONS

The 1993 Avionics IRAT determined that the V-22 avionics and VMS represented a medium risk because the program was lacking critical skills and sound systems engineering and software engineering processes. An important finding of the 1993 IRAT was that they believed the deficiencies they observed in the contractor's systems and software development processes were sufficient to raise concern over the safety of the VMS software. The IRAT determined that there was a potential for the program to become high risk if the program did not correct these problems.[Ref. 27]

As a result of the findings and recommendations of the IRATs discussed in Chapter III, the contractors took several key actions to reduce the risk to software development.

1. Software Engineering and Systems Engineering

BH has made significant progress toward developing a structured systems engineering and software engineering approach to the avionics system development. They have also focused substantial effort and attention toward separating requirements and design, documenting each appropriately. In addition, they have strengthened their process for tracing requirements from the top level, through allocation to hardware and software, down to test procedures for verifying that the system meets requirements. To aid in requirements tracing, both BH (JASS developer) and MM (VMS developer) are developing traceability database tools. The V-22 program has also begun documenting the requirements partitioning in functional and allocated baselines, the Software Segment Specification, and Software Requirements Specification.[Ref. 27]

To reduce integration risk, particularly in the development of the "coupled modes" software, BH has completed development of a triple lab tie-in to allow their three major development laboratories to be used together. This capability to exercise the major system components as an integrated whole in a laboratory environment will be invaluable in helping solve integration problems.[Ref. 27]

BH has also improved its software development environment. Nearly all the software developers have their own workstations. BH has also updated its suite of

software development tools and is making much better use of electronic documentation than was noted in the initial Avionics IRAT of 1993.[Ref. 27]

BH and MM have also made significant improvements in their testing approach. Specifically, they are developing a test-case archive that defines and documents all test cases and procedures so that specific tests can be retrieved and repeated as necessary for regression testing. The contractors plan to bring the test plans and procedures under configuration control to improve control and consistency in the testing process.[Ref. 27]

2. Staffing and Personnel

As discussed under key Government actions, the change that concerned the IRAT the most was the transition to Ada and the AMC. One of the major concerns was the lack of Ada experience at BH. Another concern was that the avionics system development was significantly understaffed, particularly in the software area. BH was having difficulty finding qualified staff.[Ref. 27]

BH has done well in acquiring Ada programmers through a combination of hiring, transfers, and training. In addition to hiring Ada programmers, other staff experienced in avionics systems have been added to the program. BH has also successfully integrated new people onto the team while retaining and strengthening the functional domain expertise carried over from FSD. The program now appears to have a solid core staff with beginning and intermediate level Ada experience. BH is also augmenting their Ada expertise by bringing in consulting help.[Ref. 27]

3. Software Development Process

Both BH and MM have strengthened their software development processes for the E&MD phase of the V-22 program. The contractors plan to go through a complete top-down process, from requirements flow-down through acceptance testing for the entire VMS software, including software that will not change in E&MD. The process will include tracing requirements to the software module level; performing design and code walk-throughs; and performing all unit, integration, and system tests.[Ref. 27]

4. Contractor Participation in Risk Management

One of the most important ingredients for success of the V-22 risk management program is contractor participation. The contractors' management has taken a positive attitude and supported the V-22 risk management program. Management support is essential to the success of the V-22 risk management program in all areas including software.[Ref. 30]

Another area where the risk management process has been enhanced is the corporate environment. Boeing has a corporate standard for risk management which requires all programs to have a risk management plan in place [Ref. 36]. The standard emphasizes the early identification and control of program risks as fundamental Boeing management objectives [Ref. 41]. The standard has been used on other programs at Boeing including AWACS, F-22, and the commercial jet, 777 [Ref. 36].

The BB program office serves as the contractor focal point for risk management and provides a co-chair for the RCB as directed in the V-22 program procedure [Ref. 32].

The RCB meets every two weeks to review the status of selected risk items, obtain agreement on new risks, and determine changes in levels to existing risks.[Ref. 42]

Management support, strong corporate process, and participation in the RCB combine to enhance contractor participation in the risk management process.

5. Subcontractor Role in Risk Management

The V-22 program office cannot directly influence the subcontractors to perform risk management. Many of the current program risks lie in the area of subcontractors. Some of the subcontractors are represented on the various IPTs and participate in the formal V-22 risk management process.[Ref. 30]

In the area of software risk management, subcontractor participation is dependent on the size of the effort. In areas such as the flight control system software development, where the effort is large and complex, the prime contractor flows the requirement for software risk management down to the subcontractor level.[Ref. 36] Software risk management is addressed in the flight control system subcontractor's software development plan.

One problem encountered by the prime contractor is that many subcontractors, both small and large, do not know how to perform risk management. For example, many subcontractors know how to identify and assess risks but are unable to perform risk reduction planning or mitigation.[Ref. 36] Consequently, risks may be present at subcontractors without proper mitigation action in place to prevent the risk from becoming a problem.

F. AN EXAMPLE OF RISK MANAGEMENT

As mentioned earlier, the area that concerned the 1993 Avionics IRAT the most was the transition to Ada and the AMC. The AMC and Run-Time Operating System Program (ROSP) are derived from the LAMPS program. V-22 cost and schedule assumed that the LAMPS program stays on schedule. Through close monitoring, it was determined that schedule slips in the LAMPS program were an increasing concern to the V-22 program.

As a result of the concern about LAMPS schedule slips, a technical risk was identified that could affect schedule. Technical managers from the Government and BH scheduled and conducted a program management review with the subcontractor responsible for AMC development. The review determined that there was, in fact, a high risk that the LAMPS schedule would slip. Accordingly, the AMC/ROSP development was changed to a high risk status. As such, it was reported to PMA-275D, the V-22 program office risk management focal point for entry into the risk database.[Ref. 39]

A risk abatement plan for the AMC/ROSP was developed by the Avionics AIT in accordance with the V-22 risk management procedure. The plan is shown in Appendix C. A risk reduction profile was developed with milestones in order to monitor the progress toward risk reduction. The profile includes such items as subassembly, software, and end item deliveries. The complete risk reduction profile is shown in Appendix E.

The following additional action was taken by the V-22 program office to reduce the risk on the AMC/ROSP.

- A trade study was performed to search for an alternate AMC vendor.

- The V-22 program office coordinated with the LAMPS program to maintain schedule and minimize impacts to the V-22 program.

- Monthly in-process reviews were scheduled and conducted with the contractors.

- The V-22 program office worked with Bell-Boeing and encouraged the subcontractor to reorganize to better meet the needs of the V-22 program. The reorganization resulted in additional emphasis on the V-22 program by the subcontractor. Bell-Boeing also provided on-site support at the subcontractor in order to more closely manage AMC development and enhance communication.[Ref. 37]

This example is typical of how computer software and hardware risks are identified, assessed, tracked, and mitigated in the V-22 program.

G. SUMMARY

This chapter has examined the V-22 risk management process as it has evolved in the E&MD phase of the acquisition process. V-22 program management sold this phase of the acquisition process as a phase that would produce production representative aircraft. Because of the huge investment of resources in the FSD phase as well as planned investment in E&MD, program management realized that it was imperative to have an aggressive risk management program in order to ensure program success.[Ref. 31] Also, high program visibility has dictated that all risks, including software, be managed in a formal, systematic, and disciplined manner.

The V-22 risk management program put into place is a well documented process whereby risks are systematically identified, addressed, and mitigated or eliminated. The management philosophy that software development should automatically be considered a

moderate risk has led to added emphasis on software risk management as a part of the overall risk management process [Ref. 31].

This chapter has attempted to highlight the importance of software risk management in the V-22 program. It has also highlighted some of the factors that have affected software risk management in the program. Key Government and contractor actions taken to reduce software development risks were discussed. Finally, an example of risk management with respect to computer software and hardware was provided along with the action taken to mitigate the risk.

The next chapter will provide an analysis of the factors affecting software risk management. It will also analyze the techniques used to manage software risk in the V-22 program. Lessons learned that can be applied to other programs will also be identified.

V. ANALYSIS AND LESSONS LEARNED

A. INTRODUCTION

Chapter IV described the risk management process for the V-22 program as it has evolved in the E&MD phase of the acquisition process. The risk management process, which was characterized as formal, systematic, and disciplined, includes software risk management as an integral part of the overall process.

As of 23 August 1995, there were 117 open risks in the V-22 risk database. Of these 117 risks, 11 were considered high. Of the 11 open items considered high risk, seven were schedule risks and four were technical risks. Counting closed items, a total of 390 risks had been identified and listed in the V-22 risk database.[Ref. 42] These figures are indicative of an environment where risks are aggressively identified and freely communicated. A well documented risk management process, which includes software risk management, is in place where risks are assessed, analyzed, and either mitigated or eliminated.

This chapter analyzes factors that have had a significant effect on risk management, and specifically software risk management, in the V-22 program. Techniques such as use of the IPT process and use of software metrics will also be analyzed. The goal is to analyze these factors and techniques for their general application to software management problems during the acquisition process. Following the analysis, lessons learned that can be applied to other programs will be identified.

B. ANALYSIS

This analysis will be structured around key factors that have affected software risk management on the V-22 program. Techniques used to manage software risk will also be analyzed.

1. Management Attention

The GAO has reported a wide variety of software development problems which plague DOD acquisition programs. One of the more significant problems identified was a lack of management attention.[Ref. 10] The V-22 program has definitely not suffered from a lack of management attention.

From the early stages of the E&MD phase, management attention has been clearly focused on software development as a risk to the V-22 program. The most notable attention with respect to software development came from the PEO with cognizance over the program. The PEO was well aware of shortfalls in the software development process during FSD. These shortfalls were chronicled in GAO reports [Refs. 24 and 25]. The PEO focused management attention on software development risk early in E&MD by chartering the first IRAT in May 1993, to identify avionics and software related risk areas as discussed in Chapter III.

As a result of early, pro-active involvement by the PEO, risks were identified early in the E&MD phase while there was ample time available to take action to reduce the risk. Also, at the time of the first IRAT in May 1993, the E&MD contract was not finalized.

This fact meant that the IRAT could have an impact on what was put into the contract as far as measures to reduce software development risk.

Continued PEO focus was maintained by the subsequent chartering of three additional IRATs as discussed in Chapter III. The IRATs were concerned with avionics software as well as flight control system software. These IRATs have been successful in identifying software development risks and offering general strategies for risk mitigation and improvement. Rather than oversight, these IRATs should be viewed as a mechanism which provided valuable assistance to the program management team of the V-22 program.

In addition to management attention at the PEO level, attention to software development risk has also been focused at the PM level. Program management has taken the approach in E&MD that all risks including software will be aggressively identified, assessed, and eliminated or mitigated. The attitude of program management is that software development should never be considered less than a moderate risk. In the V-22 program, software development is automatically considered a moderate risk and is managed accordingly. Program management also realizes the importance of having someone who is "smart" (technically competent) in the area of software to manage software development in the program. Software development is currently briefed as a moderate risk at higher level briefings given by program management.[Ref. 31]

The Deputy Program Manager for Production (PMA-275D) realized the need for a formalized risk management program early in the E&MD phase. As a result PMA-275D took the lead in developing a formal, documented risk management program.[Ref. 31]

PMA-275D has become the focal point for the risk management process in the V-22 program office. PMA-275D holds the title "Risk Manager" in addition to being the Deputy Program Manager for Production. PMA-275D implements the risk management process in the V-22 program. Having a focal point for the risk management process in the program office is essential to ensuring the maintenance of day to day management attention.

Management attention at all levels in the V-22 program has ensured that the proper focus has been maintained on software development as a risk thus far in the E&MD phase.

2. Personnel Experience/Competence

Program management emphasized the importance of getting someone who is technically competent to manage the software development process. In the E&MD phase the V-22 program has the good fortune thus far of having experienced, technically competent personnel to handle the various aspects of software development. The ASPE's qualifications were noted in Chapter IV. They included an undergraduate degree in computer science with a minor in electrical engineering; software experience in the civilian sector; and management level experience with the Government on other avionics development programs.

In addition to staffing the avionics area, other areas such as field activities supporting the VMS have been staffed with experienced personnel. Program management expressed confidence in the technical expertise and management ability of the V-22 personnel who are involved in software development. Program management also

expressed confidence in the risk management ability of the Government software experts working on the V-22 program.[Ref. 31]

Program management also has confidence in the contractor's managers in the areas of avionics and the flight control system. The contractor's managers in these areas have taken an aggressive approach to identifying and addressing software development risks [Ref. 35]. They have also put in place a good strategy, including the extensive use of metrics, for identifying and tracking potential software development risks.[Ref. 31]

Personnel who are experienced and competent in software development are essential to the success of any software intensive program. The V-22 is no exception.

3. Contract Type

Contract type has proven to be an important factor in the implementation of risk management in the V-22 program. As discussed in Chapter IV, the type of contract used in FSD was a fixed-price type contract. With this type of contract there was little opportunity for the Government to influence software development. The 1993 Flight Control Software Development Assessment Team commented that the type of contract used in FSD led the contractor to take short cuts in certain areas related to software development such as software engineering, systems engineering, and validation testing. These short cuts resulted in increased risk to the V-22 program.[Ref. 28]

In May 1994, the E&MD contract was definitized as a CPAF contract. This type of contract gives the V-22 program the flexibility to provide incentives in certain areas where additional emphasis is desired from the contractor. The area of risk management is

an award fee criterion. An award fee board meets periodically to review the contractor's performance. The amount awarded is based on the contractor's performance in relation to the award fee criteria.

The type of contract used has been instrumental in providing motivation for the contractor to demonstrate an active risk management program that identifies areas of risk and provides for corrective action.

4. Analysis and Integration Teams (AITs) and Integrated Product Teams (IPTs) and Their Roles

The V-22 program is structured with AITs and IPTs. The AITs consist of IPTs and are assigned around functional disciplines. Appendix D provides a description of how the AITs and IPTs operate and how they are structured.

From a software risk management perspective the AIT/IPT structure has a distinct advantage. Every computer software and hardware configuration item on the V-22 aircraft falls under the purview of an IPT for management purposes. The Government has membership on and participates in all IPTs. The primary method for risk identification is through the IPT process. Therefore, when a risk is identified in an IPT meeting the Government has immediate visibility. Early visibility allows appropriate corrective action to be initiated before the risk becomes a problem.

While the IPT structure is good for risk management it can have its drawbacks, especially in the early stages of implementation. The 1993 Avionics IRAT noted that the V-22 program was at an early point in the transition to an IPT structure. Because roles and responsibilities of the Government IPT members were not clearly identified, the IRAT

found that the management process was very immature and disorganized.[Ref. 27] Since 1993, the program management team has clarified and communicated the roles and responsibilities of the Government IPT representatives. For IPTs to be successful, their roles and responsibilities must be clearly defined.

With respect to risk management the reason the IPT structure has been successful is that the IPTs and AITs have clearly defined roles and responsibilities. These roles and responsibilities are formally documented in the program procedure for risk management [Ref. 32].

This research has shown that the IPT structure can be effective when properly organized. Also, it is imperative that roles and responsibilities be clearly defined. In the V-22 program the IPT structure facilitates the early identification of computer software and hardware risks. Early identification of risks allows mitigation actions to be put in place before the risks develop into the "software disasters" described in Chapter II.

5. Use of Software Metrics as a Risk Management Tool

The V-22 program office believes that software metrics are essential management indicators necessary to track software development [Ref. 23]. Software metrics provide measures of the software process, its products, and related resources. On a monthly basis, data for the metrics are collected by the V-22 contractor avionics software manager. The metrics are used to generate software management indicators (SMIs). The SMIs report status compared to plans, schedules, and allocations. Also, the SMIs provide insight into the quality level of products and the effectiveness of the software development process.

This comparison of actual results versus planned results indicates either success with carrying out plans or indicates potential problems (risks) due to unforeseen circumstances or changing needs. Indicator data can provide a basis for timely re-planning and corrective action.[Ref. 38]

Metrics are provided in the SDSR and are contractually required. The SDSR is generated on a quarterly basis and made available to management personnel, the contractor project office, Defense Plant Representative Office, and the V-22 program office.[Ref. 38] The metrics provided in the SDSR are listed below.

- Memory Resource Utilization. This metric represents the amount of memory used (both program memory and storage memory) as a percentage of the total deliverable memory.

- Processing Time Utilization. This metric represents the worst case percent usage for a major processing frame and the worst case percent usage for all minor frames.

- High Order Language (HOL). This metric represents the percent of the total object code generated by the compilation of HOL statements.

- Software Designed. This metric represents the number of computer software units (CSUs) designed as a percent of the total number of units required to satisfy the functional requirements.

- Software Coded. This metric represents the number of CSUs coded as a percent of the total number of units required to satisfy the functional requirements.

- Software Units Tested. This metric represents the number of CSUs successfully tested as a percent of the total number of units required to satisfy the functional requirements.

- Closed Problem/Change Report (PCR). This metric represents the closed PCRs as a percent of the total number of PCRs.

- Functional Requirements Verified. This metric represents the number of functional requirements fully verified as a percent of the total functional requirements in the Software Requirements Specification.

- Input/Output (I/O) Utilization. This metric represents the I/O used, both hardware and software addressable I/O units, as a percentage of the total I/O available. An I/O unit is a single discrete analog path.[Ref. 23]

In addition to the SDSR, various other reports containing metrics are submitted by the contractor. AIT and IPT meeting minutes also contain metrics and are provided monthly from the contractor to the V-22 program office.

As E&MD has progressed, the need for additional metrics was identified. These additional metrics were developed using a Boeing Defense & Space Group EOI. One of the additional metric indicators developed was the stop light chart which was discussed in Chapter IV. The stop light charts are supported by lower level metrics and are generated using a process to remove as much subjectivity as possible. The stop light charts provide a monthly software assessment. The primary benefit from the use of the stop light charts is that they provide a single top level assessment of the state of health of software development. This metric, while not contractually required, is delivered monthly for evaluation.[Ref. 23] The fact that the stop light chart is provided by the contractor even though it is not contractually required is indicative of the contractor's willingness to work with the program office to provide needed information.

Software metrics are a valuable tool to both the contractor and the V-22 program office. Indicator data, including trend analysis, is used at a detailed level by engineering and supervisory personnel at both the contractor and the V-22 program office. An

important point is that any one indicator only provides insight into a singular aspect of a process or a product. Therefore, indicators and related information are viewed collectively to determine the overall status of software development.[Ref. 38]

In the V-22 program, metrics allow managers to plan and manage the process of software development, as well as control the use of resources. Metrics also indicate where improvements can be made and allow management to measure how well the improvements add to the process.

Software metrics have also been used by the V-22 program office as an important software risk management tool. The SMIs and stop light charts are continually analyzed by the V-22 program office to determine the health of software development. The SMIs not only provide valuable data on past and current performance but are used to identify trends which indicate increased risk. The stop light chart is an excellent technique for summarizing lower level metrics.

This research has revealed that software metrics are, in fact, essential to the software risk management process in the V-22 program. Also, contractor cooperation has proven to be a factor in providing additional metrics as the E&MD phase has progressed. The research also revealed that the cost of acquiring software metrics is conservatively estimated at $75 K per year [Ref. 43]. Although formal analysis was not conducted, the Deputy Program Manager stated that the benefit of acquiring software metrics outweighs the cost [Ref. 35].

6. Contractor Improvement Efforts

Chapter III discussed the important role the IRATs played in the risk management process. The IRATs focused on the JASS and flight control system CSCIs. Several areas of concern related to contractor software development were identified early in the E&MD phase. These areas of concern, if not addressed by the contractor, would lead to increased risk in the E&MD phase of the program.

The major areas of concern were in the areas of software engineering and systems engineering. Software development processes, with respect to requirements traceability and testing, were also areas of concern. Another area of concern was the transition to Ada and the AMC for the JASS CSCI. The reason for concern was the lack of contractor staff experienced in Ada development and the immaturity of the contractor software development process.

Chapter IV identified numerous actions the contractors have taken to reduce the risk of software development in the E&MD phase of the program in the areas of avionics and the flight control system. The most important improvement was that Boeing had made significant progress toward developing a structured systems engineering and software engineering approach to the avionics system development.[Ref. 27] Other improvements were made in the areas of testing, integration, staffing, and software development environment.[Ref. 27]

These improvements, which were driven by the recommendations of the IRAT and focused management attention, have significantly reduced the risk of software development in the E&MD phase of the V-22 program.

7. Contractor Participation in Risk Management

The prime contractor team, Bell-Boeing, has actively participated in the risk management process throughout the E&MD phase. Also, Bell-Boeing management has been supportive of the V-22 risk management process. Prime contractor management support is deemed essential by the V-22 program office for successful implementation of the risk management process.[Ref. 30]

8. The V-22 Risk Management Environment

It is useful to summarize what this research has found to be the most important aspects of the V-22 risk management environment.

- Risk Management Philosophy. The V-22 risk management philosophy is to create an open, honest, risk aware culture in which risk management is considered to be a normal, healthy aspect of overall program management. The V-22 program has, in fact, created an environment where this philosophy is put into action.

- Structured and Documented Process. The risk management process is documented in a formal, written procedure that is disseminated throughout the V-22 program structure.

- Risk Management Control Board (RCB). The RCB is an integrated management team consisting of representatives from the customer and Bell-Boeing. It is responsible for overseeing the risk management process for the program and helps to ensure contractor participation.

- Risk Database. A common risk database acknowledged by both the customer and contractor. The database contains all risks, both open and closed.

- Communication. There are no barriers to communication within the program structure. The IPT process is the normal means of communicating risk information. However, both formal in informal means are used to communicate risk information.

C. LESSONS LEARNED

This section presents a list of software risk management and risk management lessons learned that were generalized from the V-22 case analysis.

- PEOs can assist program managers of software intensive programs by conducting independent risk assessments. The goal of the assessments should be to assist the PM in identifying software related risks. These assessments should be conducted early enough in the acquisition process so that risks can be identified and appropriate action taken.

- PMs should always consider software development at least a moderate risk. Many PMs have not had extensive training in the area of software development risks. Therefore, PMs may tend to consider software development as low risk or ignore it altogether. This ignorance of software can lead to disaster. In the V-22 program, the program management attitude is to consider all software development as at least a moderate risk and find someone who is "smart" on software to manage it.

- Have a program office focal point for risk management. Having a focal point for the risk management process in the program office is essential to ensuring day to day management attention.

- PMs must find someone who is experienced and technically competent to manage software development. The V-22 program has the good fortune to have experienced, technically competent personnel to manage software development.

- When contractor performance is required in a certain area, use a CPAF contract. The CPAF contract gives the program manager the flexibility to provide incentives for the contractor to perform in desired areas. In the V-22 program, risk management is a major award fee criterion.

- Make risk management fit the program structure. The V-22 program structure uses IPTs and AITs. The risk management process has been effectively tailored to fit into the IPT/AIT program structure. The IPT process is the primary method for risk identification.

- Ensure risk management roles and responsibilities are clearly understood by all involved. The formally documented V-22 risk management procedure clearly spells out the roles of IPTs, AITs, contractors, and the RCB.

- Software metrics should be used as a risk management tool. Metrics can be used not only to determine past performance but can also be used to predict future performance. The V-22 program makes extensive use of metrics to identify risks. The program office should work with the contractor to tailor the metrics package to the needs of the program.

- Contractor participation in risk management is essential. The contractor team, Bell-Boeing, has fully supported the V-22 risk management process.

- Risk management is manpower intensive. In the V-22 program many people are involved in the risk management process. Several people are involved on an almost full-time basis and many are involved on a part-time basis.

- A risk database is essential for the tracking risks.

- In large programs an integrated risk management control board with representatives from the program office and the contractor should be formed. The board serves the valuable purpose of overseeing the risk management process. It also serves the secondary purpose of ensuring contractor participation in the risk management process.

- There should be no barriers to communication of risk. In the V-22 program both formal and informal methods are use to communicate risks.

- Software risk management should be part of the overall risk management process and not a stand alone subsystem used only by software risk managers. Software risk issues should be reported in enough granularity to meaningfully identify the specific software risk.

D. SUMMARY

Many factors have influenced software risk management and risk management in the V-22 program. This chapter has analyzed the key factors that have affected software risk management as well as risk management on the V-22 program. It has also analyzed

some of the techniques to manage software risk such as metrics and the IPT structure. Finally, lessons learned which can be applied to other programs were provided.

The next chapter will provide conclusions. It will also provide a set of recommendations related to the lessons learned in this chapter.

VI. CONCLUSIONS AND RECOMMENDATIONS

This chapter will begin by presenting conclusions to the research effort. The conclusions will be followed by a set of recommendations which can be considered as possible ways to improve the acquisition of software intensive systems by DOD. These recommendations will mainly be aimed at what can be done to reduce the risk inherent in software development. The chapter will also include answers to the thesis questions used in this research effort. A recommendation for further study will be included at the end of the chapter.

A. CONCLUSIONS

This thesis began by emphasizing the important role that software plays in DOD weapon systems. Many of our key weapon systems are completely dependent on software to function properly [Ref. 1]. This software is extremely challenging to develop and has become a major source of problems in the system acquisition field. Program offices for software intensive weapon systems are facing the difficult task of managing software development risk.

Managing software risk means being able to identify, address, and eliminate software risk items before they become either threats to successful operation or major sources of software rework. Software risk management is important primarily because it helps people avoid disasters, avoid rework, avoid overkill, and stimulate win-win situations on projects involving software.[Ref. 2]

In the DOD acquisition process, risk management is required by policy. Software risk management should be an integral part of risk management for all software intensive weapon system acquisitions. However, recent estimates show that seven out of 10 major weapon systems currently in development are encountering software problems [Ref. 44]. Many of these problems could likely have been avoided had there been proper emphasis on software risk management.

The V-22 case studied in this paper presents an example of a successful risk management process which includes software risk management. As the V-22 program moved into the E&MD phase of the acquisition process, program management realized the need for a formal risk management program which aggressively identifies and mitigates or eliminates all risks, including software development risks.

The V-22 program has effectively implemented the software risk management process within the IPT/AIT program structure. Roles and responsibilities are clearly defined for software risk management. The software risk management process has been enhanced by the use of carefully tailored metrics which aid in the identification of risk. As the program has progressed in the E&MD phase, additional metrics were identified by the Government. The contractor cooperated by providing these additional metrics.

Many factors have combined to enhance the software risk management environment in the V-22 program. Management attention at the PEO(A) and PM levels provided the stimulus for increased attention to software development risk. This attention led to the IRATs which identified several software related risks. These risks were identified early

enough so that action could be taken to reduce or eliminate the risks before a software disaster occurred. Other important factors are: personnel experience/competence, contract type, and contractor participation in risk management.

The V-22 risk management process is formal and systematic. It is applied in a disciplined manner. Although it is fairly early in the E&MD phase, the action taken to reduce software development risk has had a tremendous impact. Several major risk areas have been identified, and action has been taken to eliminate or mitigate the risk. It remains to be seen how effective these actions will prove to be. First flight of aircraft number seven, the first E&MD aircraft, is scheduled to occur in December 1996. If this flight occurs as scheduled, with planned software functionality, then the actions taken will have been successful.

The lessons learned in the V-22 case can be applied to other software intensive systems in DOD. Although the V-22 is an aircraft development program, the majority of learning points involved in the software risk management process of the V-22 program are common to the DOD procurement environment. Therefore, the lessons learned can readily be generalized to a large population of software intensive systems being developed by DOD.

B. RECOMMENDATIONS

The following recommendations are a result of this research. These recommendations are aimed at managers within the acquisition hierarchy: PMs, PEOs, and component

acquisition executives who are in the best position to ensure that software risk management is implemented.

1. Create an Environment Where Risks are Freely Communicated

The PM must create an environment where risks are freely communicated. The V-22 risk management philosophy is to create an open, honest, risk-aware culture in which risk management is considered to be a normal, healthy aspect of overall program management [Ref. 32]. There must be no barriers to the communication of risk within the program management structure. Likewise, the PM should feel free to communicate program risks to the PEO. This type of environment is required in order for risk management to be successful.

2. PEOs Provide Assistance to PMs in the Area of Software Risk Management

PEOs are in a position to assist PMs in a number of ways. In the V-22 program the PEO(A) chartered several IRATs to assist the PM in identifying software related risk areas. These IRATs identified several areas of software related risk which, if not corrected, would lead to future problems. PEOs could assist PMs by having periodic risk assessments conducted on software intensive programs. The purpose of such assessments would be to assist PMs in identifying software related risk areas and to provide recommendations which would mitigate or eliminate the risk(s).

3. Use CPAF Contracts to Motivate Contractors to Perform Risk Management

CPAF contracts allow the PM to provide an incentive to the contractor(s) in certain areas where additional emphasis is desired. In the V-22 program, an award fee criterion is risk management. Having risk management as an award fee criterion is crucial in getting the contractor to participate in the risk management process.

4. Use Software Metrics as a Risk Management Tool

Metrics can be used not only to ascertain past and current performance but can also be used to predict future performance. In the V-22 program a variety of metrics are used to determine the status of software development. These metrics are continually analyzed in order to identify potential problem areas.

Through cooperation with the contractor a carefully tailored set of software metrics should be devised for every software intensive program. These metrics should be contractually required and fit the needs of the program. Metrics are by no means a "silver bullet". Metrics are just one of many tools that should be used to enhance the software risk management process.

5. Assess Contractor Capability for Software Development Prior to Contract Award

The process of assessing a contractor's software development capability was discussed in Chapter II. The SEI developed a means whereby the software process maturity of contractors could be evaluated. A developer is assessed, evaluated, its strengths and weaknesses defined, and a program of process maturity improvement is

established [Ref. 17:p. 4.15-3]. A major advantage which DOD derives from the application of the process maturity evaluation is the identification of program risks due to poor software quality products or immature development processes.

Regardless of the method used, there should be an effort to assess the contractor's capability for software development prior to contract award for software intensive weapon systems. Software capability evaluations can also be applied during the contract as a contract monitoring process. Using these evaluations offers the Government an opportunity to minimize the risk and uncertainty in software development when dealing with a particular contractor or development agency.

6. Ensure Personnel Competency in Key Positions

Most PMs have not received extensive training in the area of software development. The development of quality software is a complex, highly technical process. Therefore, the PM must have someone on the staff who is competent in the area of software development. V-22 program management emphasized the importance of having someone who is "smart" in software to manage software development for the Government [Ref. 31]

7. Ensure Roles of Risk Management Participants are Clearly Understood

This recommendation is particularly important for major programs where the program structure is large and many organizations are involved in addition to the program office. A good method to inform participants of their roles is to document their roles in a formal procedure and disseminate that procedure throughout the program structure. The

V-22 program has a written procedure for risk management. In this procedure the roles of all participants in the risk management process are clearly defined. Therefore, there is no confusion over the roles of risk management participants.

C. ANSWERS TO THESIS QUESTIONS

1. How is software risk management addressed on the V-22 aircraft program?

Software risk management can be addressed in a number of different ways. As stated in Chapter IV, there are no "cookbook" solutions to risk management. The V-22 program has a formal, systematic, and disciplined risk management process which includes software risk management. All risks, including software related risks, are aggressively identified, assessed, and mitigated or eliminated.

Software risk management is addressed in a variety of ways in the V-22 program. This process can be applied to other programs as well.

The general concepts developed from studying the V-22 case are as follows:

- Management attention at the PEO and PM level should focus on the risk of software development in software intensive weapon system acquisitions.

- The highly complex nature of software development dictates that competent, technically proficient Government personnel be assigned to manage software development on software intensive programs.

- The type of contract can have an impact on the level of contractor participation in risk management. The contract can provide an incentive for the contractor to participate in risk management.

- The software risk management process must fit into the overall risk management process as well as the program structure. In the V-22 program software risk management has been effectively implemented within the AIT/IPT program structure.

- Software metrics should be used as a risk management tool.
- Contractor participation in risk management is crucial and should be actively cultivated.

2. What are the computer software components associated with the V-22?

The CSCIs for the V-22 were identified in Chapter III. A detailed description of the CSCIs is provided in Appendix B.

3. What are the computer software components that have posed the greatest risk in the current phase of the acquisition process? What is being done to reduce the risk of software development for these software components?

The CSCIs that have posed the greatest risk in E&MD are the avionics (JASS) and flight control system (VMS).

The IRATs discussed in Chapter III were chartered to identify software related areas of risk with respect to the avionics and flight control system CSCIs. The IRATs identified several areas of risk for both of these CSCIs. Chapter IV described key Government and contractor actions that are being taken to reduce the software development risks for the avionics and flight control system CSCIs.

In addition to the IRATs the normal process for risk identification has been used to identify other computer software and hardware related risk areas. An example of one of these risks was provided in Chapter IV along with the mitigation action being taken to reduce the risk.

4. What risk management techniques were used in the past? Was software included in the risk management process?

In the FSD phase of the program, informal risk management techniques were used [Ref. 31]. Due to the informal nature of risk management in the FSD phase, as well as the time lapse since FSD, the author was unable to determine if software risk management was included in the risk management process during FSD.

5. If a formal risk management process was not used, what problems occurred in software development that may have been averted had there been a formal software risk management program?

As noted earlier, the risk management process was informal during the FSD phase of the program. The IRATs discussed in Chapter III noted many areas of concern with respect to the way software development was conducted in the FSD phase for the avionics and flight control systems. Some of the areas with serious shortcomings in FSD were systems engineering, software engineering, and testing and integration.

It is difficult to say which, if any, of the concerns identified in the IRATs could have been avoided had there been a formal risk management process in FSD. Certainly, some of the potential problems could have been identified had there been a formal process in place for software risk management. However, due to the fixed price nature of the contract in FSD, as well as the budget cutting environment of the late 1980s, it is unlikely that adequate resources could have been devoted to risk mitigation or elimination even if all of the software related risks had been identified. Also, it must be remembered that prior to 1989, there were very few sources of information that described software risks or how to deal with them [Ref. 16].

The program is now well into the E&MD phase of the acquisition process and has a formal risk management program which includes software. This environment has allowed the aggressive identification of software related risks. So far no software related problems have occurred in E&MD that were not previously identified as risks.

6. What are the lessons learned using software risk management on the V-22 program?

The lessons learned were identified in Chapter V.

D. RECOMMENDATION FOR FUTURE STUDY

More studies should be conducted in the area of software risk management. There are many other software intensive weapon system acquisitions in DOD. No doubt many of these other programs have devised innovative methods for managing the risks inherent in the development of complex software. The acquisition community could benefit from case studies of software risk management. Perhaps common software risk management techniques from several successful programs could be combined to create a paradigm for software risk management throughout DOD.

APPENDIX A. THE SEI SOFTWARE RISK EVALUATION (SRE) PROCESS

The main purpose of this appendix is to provide the reader with some background information on the SRE process. A secondary purpose is to relate the author's experiences and insight as a result of having participated in the execution phase of an SRE.

As discussed in Chapter II, one of the services that the SEI provides is the SRE. The SRE method is used for identifying, analyzing, communicating, and mitigating software risks. The SRE method is intended to be used by decision makers for managing the software risks of software intensive programs and projects. The SRE method facilitates the mitigation of software risks for managers.[Ref. 45]

The SEI has produced a technical report [Ref. 45] that provides a high level description of the method. The long term goals of the SEI with respect to the SRE method are to ensure that:

- The method is defined and can be applied in a systematic, disciplined, and efficient manner.
- For a specific program or project, at any given instance of its application, there is uniformity in the outcome of the risk findings and mitigation.
- It is flexible enough to be used in different situations and phases of the life cycle, including acquisition and maintenance.[Ref. 45]

The SRE is implemented in four phases: commitment, preparation, execution, and mitigation.

- Commitment consists of activities to establish the need, identify program or project goals, and obtain agreements for the SRE.

- Preparation consists of planning and coordination activities that are performed prior to the site visit for executing the SRE.

- Execution consists of activities to implement the SRE functions during a site visit to the location of the target program.

- Mitigation consists of activities to mitigate the risks that were identified during the software risk evaluation.[Ref. 45]

The SRE can be applied in different situations and within different environments. The author was a member of an SEI team that was recruited by a Government program office. The program manager requested an independent evaluation of the software development risks facing the organization.

Of the four activities mentioned above, the author actively participated in the execution phase of the SRE. The SRE team consisted of seven members distributed as follows: three from SEI, two from an independent organization, one from the program office, and the author. The team members performed various roles during the SRE. Since the author participated in the execution phase, that phase of the SRE will be addressed here.

The execution phase consists of those activities that are performed during a site visit of the SRE team to the location of the target program. SREs are tailored to an organization's needs, and each SRE may vary slightly depending on the organization. This particular SRE execution phase lasted one week and consisted of the following activities: site orientation/program briefing, interview sessions, analysis sessions, risk data consolidation, and a results briefing.

The purpose of the site orientation is for all individuals participating in the execution phase activities to be aware of:

- the objectives of performing the evaluation
- the implementation phases of the SRE
- the individual's role in the implementation phases.[Ref. 45]

The purpose of the program briefing is for knowledgeable representatives to present a summary of the organization's structure, context, and technical aspects of the target program.[Ref. 45]

The purpose of the interview sessions is to perform risk detection and specification. Each interview session begins with the interviewer using an introductory script followed by questions from the taxonomy-based questionnaire (TBQ). The TBQ is a tool used to identify software risks in a program. The purpose of this tool is to ensure coverage of all potential risk areas by asking questions at a detailed attribute level of the software risk taxonomy. The SEI software risk taxonomy provides a basis for organizing and studying various aspects of software risks in a program.[Ref. 45]

The interview process is designed to facilitate the detection of risks on the basis of the participants' discussion rather than by rigidly following the structure of the TBQ. The participants are encouraged to follow any thread of discussion or thought as long as they are objectively discussing potential risks when responding to questions, or to subsequent follow on probes, or to cues, or begin to discuss among themselves.[Ref. 45] In this

particular SRE there were seven interview groups with a total of 35 individuals. The groups consisted of a variety of personnel including management and technical personnel.

When a risk is identified the risk recorder documents the risk. The risk recorder uses the wording of the respondents, and if the meaning is not clear will clarify the statement before documenting it. The risk recorder or another designated person is responsible for entering the risks into an automated analysis tool.[Ref. 45]

A session recorder takes notes on the context and other pertinent discussions during the risk detection activities. Other SRE team members also take notes to ensure the following:

- Any potential risk, issue, or concern that was raised by an interviewee is not overlooked.
- The source of the risk is clearly identifiable and can be tagged to a category in the SEI taxonomy of software development risks.
- Sufficient information is available for the team to make an objective assessment of each risk that was detected.[Ref. 45]

The purpose of the analysis session is to complete the functions of risk specification and risk assessment. The analysis session is performed by the SRE team. The session is performed immediately after each interview session when the context of the risk is still fresh in the minds of the SRE team members. Each team member is provided a copy of the recorded risks. The SRE team discusses only those risks where the wording of the risk statements is of concern and may need to be changed. The team members individually score each risk using the selected assessment mechanism. The team then discusses those risks that have a significant deviation in their scores and reaches consensus. During the

analysis session each risk will also be tagged to a taxonomy group; that is, it will be tagged as belonging to a specific class-element-attribute in the software risk taxonomy. During this session the risk recorder or a designated person enters the risk statement corrections, risk assessment scores, and source of risk categories into the automated analysis tool.[Ref. 45]

The purpose of the consolidation session is to perform the risk consolidation and if necessary, revise the assessments of the consolidated risks. The session is performed by the SRE team and is held after all the analysis sessions have been completed. Each SRE team member is provided with a copy of the risks from the analysis sessions sorted by their levels of magnitude within each risk category. The SRE team jointly examines the risks within each category to determine if there are candidates for consolidation. The team reaches consensus on the wording and revisits the assessment scores of the consolidated risks if necessary. A risk recorder enters the consolidated risk statements and their revised risk assessments into the automated analysis tool.[Ref. 45]

After the consolidation of risks a briefing is prepared. The contents of the briefing are a listing of the risk statements that were identified during the execution phase and sorted by their source of risk categories and levels of magnitude. Although all risk findings may be presented, the focus of the data confirmation should be on the more important ones, that is, those risks that were assessed at a high level of magnitude.[Ref. 45] In this SRE, 246 risk statements were collected and consolidated into 12 major risk source areas.

The consolidation represented those risks rated to have a high impact on program software efforts and/or high probability of occurrence.

The actual briefing is presented as the last step in the execution phase. The purpose of the briefing is to validate the risk findings with the organization's management and all individuals who participated in the execution phase activities. All individuals who were involved in the execution phase activities including the management and technical personnel associated with the target program and the SRE team members are encouraged to attend the briefing session. The briefing session provides feedback to the organization by openly communicating the risks that were found and provides an opportunity for the data gathered at the site to be validated.[Ref. 45]

It is important to remember that the author only participated in the execution phase of this particular SRE. The other phases of the process are: commitment, preparation, and mitigation.

The author actively participated in the execution phase in the roles of risk recorder, session recorder, and observer. The author also participated in the analysis and consolidation sessions. The following are the author's observations with respect to the execution phase of the SRE process.

- The process is well defined, disciplined, and systematic.
- The process can be tailored to fit different programs in different phases of acquisition.
- The process is an excellent tool for identifying software development risks. The process can and probably should be used for all software intensive DOD acquisition programs.

- The process for assessing risks is somewhat subjective and depends heavily on the expert judgment of team members.

- All software development risks may not be identified. However, the author is confident that at least 85-90% of the major risks were identified.

- Program manager commitment to the process is extremely important.

- Organizational commitment/participation is also extremely important. The right program personnel must be available to the team for the interviews at the designated time. In this case, there was 100% participation from the organization that was being evaluated. There was also a good cross-section of experience among the interviewees ranging from management to software developers.

APPENDIX B. DESCRIPTION OF V-22 COMPUTER SOFTWARE CONFIGURATION ITEMS (CSCIs) AND ASSOCIATED HARDWARE

1. Integrated Avionics System

The V-22 Integrated Avionics System (IAS) consists of hardware and software installed and integrated in the V-22 aircraft. The avionics system is comprised of dual redundant Advanced Mission Computers (AMCs) configured with a single Reduced Instruction Set Computing (RISC) Computer Module (RCM) based on a MIPS R4400 processor which perform system control, system monitoring, MIL-STD-1553B data bus control, and subsystem processing functions. The mission computers are integrated with the avionics equipment through three dual redundant MIL-STD-1553 data buses. Two of these buses allow for communication between the two AMCs and the IAS. The third bus is dedicated to communication between the two AMCs. JASS controls the IAS either through one of the data buses directly or through one of four interface units (IUs). the IUs provide data conversion between the MIL-STD-1553B formatted data to and from the mission computer and other data formats used by non MIL-STD-1553B compatible equipment.[Ref. 23]

The JASS tactical software provides integrated control and centralized data processing for the V-22 avionics system. Ada is the language used in developing JASS.[Ref. 23]

2. Digital Flight Control System (DFCS)

The V-22 DFCS consists of primary and secondary devices including electronic sensing and computing devices which in combination with the aircraft control surfaces enable the crew to control the flight path of the aircraft. It provides primary flight control system (PFCS) functions necessary for safe control of the aircraft as well as automatic flight control system (AFCS) functions required to accomplish V-22 missions. It commences with flight crew cockpit controls and extends through the surface and swashplate actuators and digital engine controls. It is fly-by-wire implementation employing triplex in-line monitored sensors and computers, and shall be two-fail/operate with respect to sensing, computing, and control valves for flight critical (PFCS) functions and one-fail/operate with respect to automatic (AFCS) functions.[Ref. 23]

The flight control computer software is implemented in one CSCI. This CSCI provides logic for input/output, control law, and built-in-test processing. The PFCS control laws perform all flight critical computations and redundancy management. The AFCS control laws include those providing stability and control augmentation and mission related selectable modes of the flight control system. The input/output (I/O) processor logic controls most of the hardware interfaces in the flight control computer. It then passes the I/O data between the interfaces and the PFCS and AFCS operational flight programs. DFCS software is coded in assembly language.[Ref. 23]

3. V-22 Mission Planning System (VMPS)

The VMPS will provide V-22 personnel with an automated mission planning system. The primary purposes of the VMPS are to allow a pilot to plan out a mission and populate the databases in the V-22 aircraft mission computer just prior to commencing mission and to provide digital map data for the on-board digital map system. The Tactical Automated Mission Planning System (TAMPS) has been designated as the Navy and Marine Corps standard mission planning system. The TAMPS is a ground based system consisting of core hardware and software that provides common mission planning data to various aircraft types in the Navy and Marine Corps. A platform specific V-22 Mission Planning Module (VMPM) is in development to operate within the TAMPS environment.[Ref. 23]

The VMPM is a software configuration item that physically runs within TAMPS. TAMPS will download data to a cartridge to upload flight and communication plans into the aircraft. The VMPM software must accept output from the core TAMPS system and provide data in a format compatible with the avionics, communications, and digital mapping systems of the V-22. VMPS software is coded in C.[Ref. 23]

4. Maintenance Data Processing System (MDPS)

The MDPS supports maintenance of the V-22 by receiving, processing, reporting, and distributing V-22 maintenance data. The primary missions of the MDPS are reading and processing data recorded on a Mission Data Loader (MDL) by a V-22 during flight, maintaining aircraft configuration and usage data, and generating reports. The MDPS also

maintains inventories of installed V-22 life-limited parts at the local organization, reports data up-line to data collection agencies, and exchanges data with other MDPS stations. The MDPS hardware consists of an 80486/66 Megahertz processor along with peripheral devices to support operation of MDPS software.[Ref. 23]

The MDPS CSCI is menu-driven. The menus will present a structured and standardized appearance that is usable by personnel with any level of computer experience. Displays other than menus will be generated and presented to the user as rapidly as possible. Data entry screens will be formatted to ease the entry of information by the operator. Display and entry of data for standard maintenance forms will duplicate the format of the paper form. Information typed by the user will be verified as it is entered. MDPS software is coded in Ada.[Ref. 23]

5. Display Electronics Unit (DEU)

The primary purpose of the V-22 DEU is to provide an integrated interface for the control and display portion of the JASS and display units. To support this interface, the DEU provides display generation processing as directed by the JASS and processing of operator control inputs received via the multi-function displays. Operator control inputs and display system status are provided for sampling by the JASS over MIL-STD-1750A processors in the DEU systems. DEU software is coded in C and Assembly.[Ref. 21]

6. Vibration, Structural Life, and Engine Diagnostic (VSLED)

The VSLED system contributes functionally to the V-22 central integrated checkout system as an aircraft health monitoring system. The VSLED system provides a

data collection, storage, and on-board analysis capability for the aircraft drive system and its associated support structure. VSLED software is coded in Jovial.[Ref. 21]

7. Interface Units

Several interface units, consisting of both hardware and software, exist to perform various functions. These interface units are the Avionics Bay Interface Unit (ABIU), Wing Interface Unit (WIU), and Nacelle Interface Units (NIUs).[Ref. 21]

The ABIU consists of hardware and software installed and integrated with the avionics and non-avionics equipment through a dual redundant MIL-STD-1553B data bus, other serial buses, and various analog and discrete signals. ABIU hardware is based on an 8086 processor and controls/monitors different subsystems or functions. ABIU firmware is coded in C and 8086 assembly.[Ref. 21]

The WIU consists of hardware and software installed and integrated with the avionics and non-avionics equipment through a dual redundant MIL-STD-1553B data bus (bus A), other serial buses, and various analog and discrete signals. WIU hardware is based on an 8086 processor. The WIU controls/monitors 10 different subsystems or functions. The WIU firmware is wiitten in C and 8086 assembly.[Ref. 21]

There are two identical NIUs on the V-22 aircraft. NIU #1 is located in the left engine nacelle and monitors/controls the left engine and associated subsystems. NIU #2 is located in the right engine nacelle and monitors/controls the right engine and associated subsystems. The NIUs consist of hardware and software installed and integrated with the avionics and non-avionics serial buses and various analog and discrete signals. The NIU

hardware is based on an 8086 processor. The NIUs control/monitor 11 different subsystems or functions. The NIU firmware is written in C and 8086 assembly.[Ref. 21]

8. JVX Simulation Support Software (JSSS)

The JSSS both simulates and stimulates actual avionics hardware for use in testing of the JASS. The JSSS performs simulation functions such as the flight simulation function which allows for control of simulated trajectories such as waypoint steering and joystick control. This function also controls the flight modeling. Other important functions are also simulated by the JSSS. JSSS software is coded in Fortran.[Ref. 23]

APPENDIX C. V-22 RISK ITEM

Date 6/19/95

ID	TITLE		
105	AMC		
ASSESSMENT	CATEGORY	PRIME LOC	RESP MGR
H	T	Phl/PMA 275	D. Moorman

- **Description of Potential Risk Area**

  –The V-22 Advanced Mission Computer (AMC) and Run-Time Operating System Program (ROSP) are derived from the LAMPS program. V-22 cost and schedule assume that LAMPS program stays on schedule.

- **Current Status**

  –Elevated to Moderate risk, RCB video 7/15/94

  –Elevated to High risk, RCB video 4/19/95

  –The development of the V-22 AMC and ROSP continues to be impacted by delays in the LAMPS program. CDI has slid delivery of H/W and S/W by approximately 6 months. There is no remaining schedule reserve in the V-22 schedules. Any additional slides in delivery will impact the content of JASS S/W for first flight.

- **Program Action If No Action Taken**

  –Less than planned S/W functionality for A/C 7 first flight

- **Action Required to Reduce Risk to Acceptable Levels**

  1) Develop fallback plans for contingency of additional delivery delays.

  2) Coordination with LAMPS program to identify impacts on V-22.

  3) Customer emphasis on LAMPS program to maintain schedule and minimize impacts to V-22.

- **Fallback Plans/Workarounds**

  Options being considered: (1) Compression of lab integration and testing. (2) Offload of testing from Boeing SIL to NAWC VAIL. (3) Hiring additional S/W designers at Boeing to minimize impact of late delivery from CDI.

- **Impact of Fallback Plans/Workarounds**

  –Additional unplanned cost

- **Closure Criteria**

  –Delivery of AMC and software in Feb 1996

- **Due Date for Next Action**

  5/17/95 CDI PMR

  5/31/95 Completion of detailed workaround/fallback plan

- **Recommendation to Management**

  –Monitor closely and maintain pressure on customer to resolve situation.

APPENDIX D. ANALYSIS AND INTEGRATION TEAMS (AITs) AND INTEGRATED PRODUCT TEAMS (IPTs)

The AITs consist of different IPTs and are designed around functional disciplines. These teams, within their own structure, link activities from all IPTs together by identifying objectives and scheduling product hand-off in a coordinated fashion. The Government and Bell-Boeing assembled representatives for each AIT. These representatives coordinate all their design issues within their assigned structure. All Government AITs are structured under the Air Vehicle AIT, which is managed by the V-22 Class Desk. The AIT structure is depicted below.[Ref.23]

- Air Vehicle
  - Avionics
  - Crew System
  - Air Frame & Systems
    - Air Frame: Forward Fuselage
Center Fuselage
Aft Fuselage
Wing
Rotor
Drive System
Propulsion
Subsystem & Integrated Wiring System
Empennage & Ramp
    - Systems
  - Vehicle Management System

V-22 IPTs are made up of Bell-Boeing and Government experienced engineers/ leaders. Each IPT operates as a miniature, self-contained program having ownership of a

specific product and responsibility for all aspects of its development. Through IPTs, the Government is involved in the design process early on.[Ref.23]

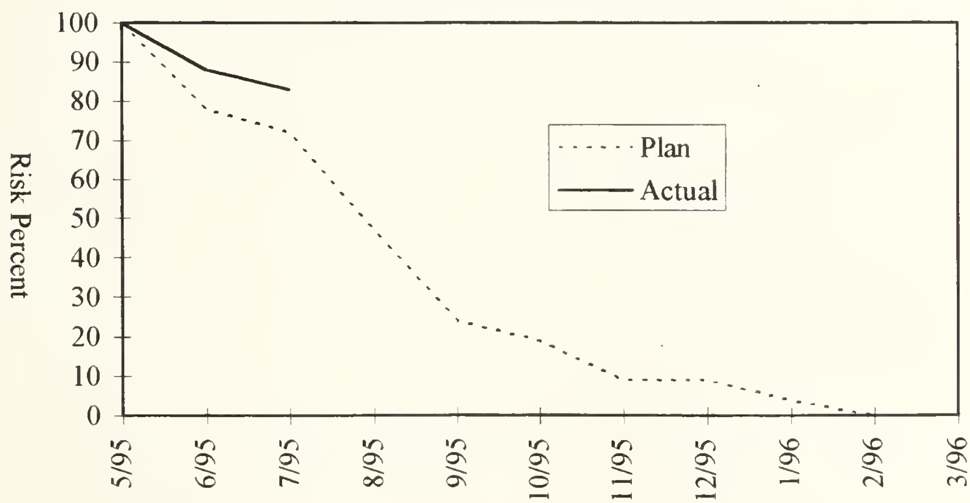
IPT leaders perform a role similar to a program manager, assuming full responsibility for delivery of a product, which meets all customer requirements, schedules, and budgets. Teams are sized small enough to enhance the working environment and internal communication. The IPTs are responsible for requirements refinements within their teams. Design changes/iterations are minimized by involving all functional disciplines (i.e., reliability and maintainability, weight, cost, manufacturing, quality assurance, material, engineering, etc.) in the initial design cycle.[Ref.23]

APPENDIX E. RISK RECUCTION PROFILE

AMC/ROSP Risk Reduction Profile

Description	Plan	Actual	Weight %
LBC-02 ASIC Release to FAB	6/5/95	6/23/95	2
MCC-02 ASIC Release to FAB	6/5/95	6/30/95	2
First ROSP w/1553 Drop	6/16/95	6/16/95	5
IOC ASIC Stress Testing Complete	6/16/95	6/20/95	3
LIFE 3.0 Prototype Delivery	6/23/95	7/5/95	5
Frist AMC Block 1 Delivery	6/27/95		5
LBC-02 ASIC Prototype Delivery	7/31/95		3
MCC-02 ASIC Prototype Delivery	7/31/95		3
Last AMC Block 1 Delivery	8/15/95		5
Benchmark Complete	6/16/95		10
ROSP Full 1553 Drop	8/16/95		10
First AMC Block 2 Delivery	9/19/95		5
Full Single Module Integration Complete	9/20/95		5
SOF Qualification Complete	9/30/95		10
ROSP Block 2 Complete	10/18/95		5
ROSP Block 2 FQT	11/17/95		10
A/C 7 Hardware & Spares	1/31/96		5
A/C 8 Hardware & Spares	2/29/96		4

AMC / ROSP Risk Reduction Profile



APPENDIX F. LIST OF PERSONNEL INTERVIEWED

1. Smith, B., Deputy for Production, V-22 Program Office, Naval Air Systems Command, Arlington, VA, Interview, 19 September 1995.

2. Giles, T., Flight Control System Support Activity Manager, Naval Air Warfare Center, Patuxent River, MD, Interview, 19 September 1995.

3. Schleicher, R., Deputy Program Manager, V-22 Program, Naval Air Systems Command, Arlington, VA, Interview, 19 September 1995.

4. Quinn, A., Defense Plant Representative Office, Philadelphia, PA, Interview, 20 September 1995.

5. Kennedy, F., V-22 Avionics System Project Engineer, Naval Air Systems Command, Arlington, VA, Interview, 21 September 1995.

6. Tolan, G., Project Lead for the V-22 Mission Planning System, Naval Air Warfare Center, Indianapolis, IN, Interview, 22 September 1995.

7. Heselton, L., Boeing Helicopter, Risk Reduction and Test Support Integrated Product Team, Philadelphia, PA, Interview by Phone, 27 October 1995.

8. Schleicher R., Deputy Program Manager, V-22 Program, Naval Air Systems Command, Arlington, VA, Interview by Phone, 7 February 1996.

9. Kennedy, F., Avionics System Project Engineer, Naval Air Systems Command, Arlington, VA, Interview by Phone, 12 February 1996.

APPENDIX G. LIST OF ACRONYMS

ABIU	Avionics Bay Interface Unit
AFCS	Automatic Flight Control System
AIT	Analysis and Integration Team
AMC	Advanced Mission Computer
ASIC	Application Specific Integrated Circuit
ASPE	Avionics Systems Project Engineer
BB	Boeing-Bell Helicopter Textron, Inc.
BH	Boeing Defense & Space Group, Helicopter Division
CDI	Computing Devices International
CMM	Capability Maturity Model
COMM	Communication
CPAF	Cost-Plus-Award-Fee Contract
CRLCMP	Computer Resources Life Cycle Management Plan
CSCI	Computer Software Configuration Item
DEU	Digital Electronics Unit
DFCS	Digital Flight Control System
DOD	Department of Defense
DSMC	Defense Systems Management College
E&MD	Engineering and Manufacturing Development
EOI	Engineering Operating Instruction

FCC	Flight Control Computer
FCS	Flight Control System
FCSDAT	Flight Control System Development Assessment Team
FCSIR	Flight Control System Integration Rig
FSD	Full-scale Development
GAO	General Accounting Office
HW	Hardware
I/O	Input/Output
IAS	Integrated Avionics System
IOC	Initial Operating Capability
IOC	Input/Output Controller
IPT	Integrated Product Team
IRAT	Independent Risk Assessment Team
IU	Interface Unit
JASS	V-22 (JVX) Applications Systems Software
JSSS	V-22 (JVX) Simulation Support Software
JVX	Joint Services Vertical Lift Aircraft
LAMPS	Light Airborne Multi-Purpose System
LBC	Local Bus Controller
LIFE	Local Interface Futurebus+ Engine
MCC	Memory Computer Controller

MCCR	Mission Critical Computer Resources
MDL	Mission Data Loader
MDPS	Maintenance Data Processing System
MIPS	Millions of Instructions Per Second
MM	Martin-Marietta
NAV	Navigation
NAWC	Naval Air Warfare Center
NIU	Nacelle Interface Unit
PEO	Program Executive Officer
PFCS	Primary Flight Control System
PM	Program Manager
PO	Project Officer
RCB	Risk Management Control Board
RCM	RISC Computer Module
RISC	Reduced Instruction Set Computing
ROSP	RCM Operating System Program
SDSR	Software Development Status Report
SEI	Software Engineering Institute
SIL	Systems Integration Laboratory
SLOC	Source Lines of Code
SMI	Software Management Indicator

SOF	Special Operations Forces
SPAT	Software Product Assessment Team
SW	Software
TAMPS	Tactical Automated Mission Planning Station
TRR	Test Readiness Review
VAIL	V-22 Avionics Integration Laboratory
VMPM	V-22 Mission Planning Module
VMPS	V-22 Mission Planning System
VMS	Vehicle Management System
VSLED	Vibration, Structural Life, and Engine Diagnostics
WIU	Wing Interface Unit

LIST OF REFERENCES

1. Cummings, T., *Corrective Software Management: The Success of the EPLRS Program*, Master's Thesis, Naval Postgraduate School, Monterey, CA, March 1994.

2. Boehm, B.W., *Tutorial: Software Risk Management*, IEEE Computer Society Press, 1989.

3. Defense Science Board, *Report of the Task Force on Military Software*. September 1987.

4. Department of Defense, *U.S. National Strategy*, 1993.

5. Defense Systems Management College, *Mission Critical Computer Resources Management Guide*, 1988.

6. United States General Accounting Office, GAO/IMTEC-92-62BR, *Embedded Computer Systems, Defense Does Not Know How Much It Spends on Software*, Briefing Report to the Chairman, Subcommittee on Research and Development, Committee on Armed Services, House of Representatives. July 1992.

7. Software Engineering Institute Technical Report CMU/SEI-90-TR-12, Carnegie Mellon University, *National Software Capacity: Near-Term Study*, by J.A.L. Siegal, S. Stewman, S. Konda, P.D. Larkey, and W.G. Wagner, May 1990.

8. United States General Accounting Office, GAO/IMTEC-90-23, *Meeting the Government's Technology Challenge: Results of GAO Symposium*. February 1990.

9. Kitfield, J., "Is Software DOD's Achilles' Heel?," *Military Forum*, July 1989.

10. United States General Accounting Office, GAO/IMTEC-93-13, *Mission Critical Systems, Defense Attempting to Address Major Challenges*, Report to the Chairman, Subcommittee on Research and Development, Committee on Armed Services, House of Representatives. November 1992.

11. Software Engineering Institute Technical Report CMU/SEI-92-TR-30, Carnegie Mellon University, *Software Development Risk: Opportunity, Not Problem*, by Roger L. Van Scoy, September 1992.

12. Defense Systems Management College, *Risk Management Concepts and Guidance*, March 1989.

13. Software Engineering Institute Technical Review '92, Carnegie Mellon University, *Software Development Risk Management: An SEI Appraisal*, by Robert J. Kirpatrick, Julie A. Walker, and Robert Firth, 1992.

14. Department of Defense Directive 5000.1, *Defense Acquisition*, 23 February 1991.

15. Department of Defense Instruction 5000.2, *Defense Management Acquisition Procedures*, 23 February 1991.

16. Jones, C., *Assessment and Control of Software Risks*, Yourdon Press, 1994.

17. Defense Systems Management College, *Program Manager's Notebook*, June 1992.

18. United States General Accounting Office, GAO/NSIAD-93-15, *Weapons Acquisition, A Rare Opportunity for Lasting Change*, Report from the Comptroller General of the United States. December 1992.

19. United States General Accounting Office, GAO/IMTEC-92-48, *Embedded Computer Systems, Significant Software Problems on C-17 Must Be Addressed*, Report to the Chairman, Subcommittee on Legislation and National Security, Committee on Government Operations, House of Representatives. May 1992.

20. *Operational Requirements Document* for Joint Multi-mission Vertical Lift Aircraft (JVMX), 4 April 1995.

21. *Cost Analysis Requirements Description for MV-22, CV-22, and HV-22,* CARD-94-04 (draft), 15 April 1994.

22. Department of Defense Inspector General Audit Report, *Review of the V-22 Aircraft Program*, no. 94-131, 14 June 1994.

23. V-22 Program Document, *Computer Resources Life Cycle Management Plan*, 25 September 1995.

24. United States General Accounting Office, GAO/NSIAD-91-45, *Naval Aviation, The V-22 Osprey -- Progress and Problems*, Report to the Ranking Minority Member, Committee on Armed Services, House of Representatives. October 1990.

25. United States General Accounting Office, GAO/NSIAD-94-44, *Navy Aviation, V-22 Development -- Schedule Extended, Performance Reduced, and Costs Increased*, Report to the Chairman, Committee on Armed Services, House of Representatives. January 1994.

26. Integrated Program Summary, Annex D, *Risk Assessment for the V-22 Program*.

27. Department of the Navy, Program Executive Officer for Air Anti-Submarine Warfare (ASW), Assault, and Special Mission Programs, *V-22 Avionics System Independent Risk Assessment Final Report*, Arlington, VA, 1 September 1994.

28. National Aeronautics and Space Administration, Ames Research Center, *Final Report, V-22 Flight Control System Software Development Assessment Team*, Moffett Field, CA, 31 August 1993.

29. Department of the Navy, Program Executive Officer for Air ASW, Assault, and Special Mission Programs, *V-22 Full Scale Development, Digital Flight Control System Software Product Assessment Final Report*, Arlington, VA, 16 December 1994.

30. Interview between Mrs. Barbara Smith, V-22 Risk Manager, and the author, 19 September 1995.

31. Interview between Mr. Ray Schleicher, Deputy Program Manager for the V-22 Program, and the author, 19 September 1995.

32. V-22 Program Policy and Procedure Number 211A, *V-22 Risk Management*, 21 October 1994.

33. *Program Executive Plan for the V-22 Engineering and Manufacturing Development Program*, 16 July 1993.

34. Briefing Charts from Briefing on *V-22 Risk Management*, 9 May 1995.

35. Telephone interview between Mr. Ray Schleicher, Deputy Program Manager for the V-22 Program, and the author, 7 February 1996.

36. Telephone interview between Mr. Les Heselton, Boeing Defense & Space Group, Helicopters Division, and the author, 27 October 1995.

37. Interview between Mr. Frank Kennedy, Avionics Systems Project Engineer for the V-22 Program, and the author, 21 September 1995.

38. *Software Development Plan* for the V-22 Application System Software, 16 September 1994.

39. Telefax from Mr. Frank Kennedy, V-22 Program Office, 14 August 1995.

40. Electronic mail message from Mr. Frank Kennedy, Avionics Systems Project Engineer for the V-22 Program, 20 October 1995.

41. Boeing Defense & Space Group Operating Procedure MA-DCT-026, *Risk Management*, 15 November 1990.

42. Meeting minutes from V-22 Risk Management Control Board meeting, 25 August 1995.

43. Telephone interview between Mr. Frank Kennedy, Avionics Systems Project Engineer for the V-22 Program, and the author, 12 February 1996.

44. United States General Accounting Office, GAO/IMTEC-90-34, *Embedded Computers, Better Focus on This Technology Could Benefit Billion Dollar Weapons Programs.* Report to the Chairman, Legislation and National Security Subcommittee, Committee on Government Operations, House of Representatives. April 1990.

45. Software Engineering Institute Technical Report CMU/SEI-94-TR-19, Carnegie Mellon University, *Software Risk Evaluation Method Version 1.0*, by Frank J. Sisti and Sujoe Joseph, December 1994.

BIBLIOGRAPHY

Boatman, J., "Bell-Boeing Prepares for V-22 Deadline," *Jane's Defence Weekly*, p. 14, 9 October 1993.

Boehm, B.W., "Software Engineering," *IEEE Transactions on Computers*, vol. C-25, no. 12, December 1976.

Boehm, B.W., "Software Risk Management: Principles and Practices," *IEEE Software*, vol. 8, iss. 1, pp. 32-41, January 1991.

Boehm, B.W., *Tutorial: Software Risk Management*, IEEE Computer Society Press, 1989.

Brill, A.E., *Techniques of EDP Project Management: A Book of Readings*, Yourdon Press, 1984.

Bunyard, J.M., "Today's Risks in Software Development -- Can They Be Significantly Reduced?" *The Journal of Defense Systems Acquisition Management*, vol. 5, no. 4, pp. 73-94, 1982.

Cost Analysis Requirements Description for MV-22, CV-22, and HV-22, CARD-94-04 (draft), 15 April 1994.

Cummings, T., *Corrective Software Management: The Success of the EPLRS Program*, Master's Thesis, Naval Postgraduate School, Monterey, CA, March 1994.

Davis, R., "Reducing Software Management Risks," *Defense Systems Management Review*, vol. I, no. 6, pp. 16-23, Defense Systems Management College, Fort Belvoir, VA, 1978.

Defense Science Board, *Report of the Task Force on Military Software*. September 1987.

Defense Systems Management College, *Mission Critical Computer Resources Management Guide*, 1988.

Defense Systems Management College, *Program Manager's Notebook*, June 1992.

Defense Systems Management College, *Risk Management Concepts and Guidance*, March 1989.

Defense Systems Management College, *Systems Engineering Management Guide*, January 1990.

DeMarco, T., *Controlling Software Projects*, Yourdon Press, 1982.

Department of Defense Directive 5000.1, *Defense Acquisition*, 23 February 1991.

Department of Defense Inspector General Audit Report, *Acquisition of the V-22 Joint Services Advanced Vertical Lift Aircraft (Osprey)*, no. 89-077, 14 June 1989.

Department of Defense Inspector General Audit Report, *Review of the V-22 Aircraft Program*, no. 94-131, 14 June 1994.

Department of Defense Instruction 5000.2, *Defense Management Acquisition Procedures*, 23 February 1991.

Department of Defense Manual 4245.7-M, *Transition From Development to Production...Solving the Risk Equation*, September 1985.

Department of Defense Manual 5000.2-M, *Defense Acquisition Management Documentation and Reports*, February 1991.

Department of Defense. *Report of the Defense Science Board Task Force on Acquiring Defense Software Commercially*. June 1994.

Department of Defense. *Software Development and Documentation*. MIL-STD-498. 5 December 1994.

Department of Defense. *Technical Report on Work Breakdown Structure Elements for Software*. 10 July 1992.

Department of Defense, *U.S. National Strategy*, 1993.

Department of Defense. *Work Breakdown Structures for Defense Material Items*. MIL-STD-881B. 25 March 1993.

Department of the Navy, NAVSO P-6071, *Best Practices - How to Avoid Surprises in the World's Most Complicated Technical Process...The Transition From Development to Production*, March 1986.

Department of the Navy, Program Executive Officer for Air Anti-submarine Warfare (ASW), Assault, and Special Mission Programs, *V-22 Avionics System Independent Risk Assessment Final Report*, Arlington, VA, 1 September 1994.

Department of the Navy, Program Executive Officer for Air ASW, Assault, and Special Mission Programs, *V-22 Full Scale Development, Digital Flight Control System Software Product Assessment Final Report*, Arlington, VA, 16 December 1994.

Ferrel, D.W., "Navy Airborne Weapon System Software Acquisition," *Defense Systems Management Review*, vol. I, no. 6, pp. 47-53, Defense Systems Management College, Fort Belvoir, VA, 1978.

Glennan, T.K., Bodilly, S.J., Camm, F., Mayer, K.R., and Webb, T.J., *Barriers to Managing Risk in Large Scale Weapon System Development Programs*, The Rand Corporation, 1994.

Holzer, R., "V-22 Earns Vital Go-ahead," *Navy Times*, vol. 43, no. 50, p. 36, 19 September 1994.

Integrated Program Summary, Annex D, *Risk Assessment for the V-22 Program*, 1994.

Johnston, J., "Commercial Techniques Trim Tiltrotor's Price Tag," *National Defense*, vol. 79, no. 503, pp. 48-49, December 1994.

Jones, C., *Assessment and Control of Software Risks*, Yourdon Press, 1994.

Kitfield, J., "Is Software DOD's Achilles' Heel?," *Military Forum*, July 1989.

National Aeronautics and Space Administration, Ames Research Center, *Final Report, V-22 Flight Control System Software Development Assessment Team,* Moffett Field, CA, 31 August 1993.

Naval Research Laboratory. *The Mudd Report: A Case Study of Navy Software Development Practices*. 21 May 1975.

Neumann, P.G., *Computer Related Risks*, ACM Press, 1995.

Office of Management and Budget, "Major System Acquisitions." *OMB Circular No. A-109 to the Heads of Executive Departments and Establishments*. 5 April 1976.

O'Brien, M.A., *The V-22 Osprey: A Case Analysis*, Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1992.

Operational Requirements Document for Joint Multi-mission Vertical Lift Aircraft (JVMX), 4 April 1995.

Program Executive Plan for the V-22 Engineering and Manufacturing Development Program, 16 July 1993.

Secretary of the Navy Instruction 5000.2A, *Implementation of Defense Acquisition Management Policies, Procedures, Documentation, and Reports*, 9 December 1992.

Secretary of the Navy Instruction 5000.32A, *Acquisition and Management Policies and Procedures for Computer Resources*, 3 May 1993.

Sedivy, D.G., *Bureaucracies at War: The V-22 Osprey Program*, Executive Research Project F37, The Industrial College of the Armed Forces, National Defense University, Fort McNair, Washington, D.C., April 1992.

Software Development Plan for the V-22 Application System Software, 16 September 1994.

Software Engineering Institute Special Report CMU/SEI-94-SR-9, Carnegie Mellon University, *Software Acquisition: A Comparison of DOD and Commercial Practices*, by Jack R. Ferguson and Michael E. DeRiso, October 1994.

Software Engineering Institute Technical Report CMU/SEI-90-TR-12, Carnegie Mellon University, *National Software Capacity: Near-Term Study*, by J.A.L. Siegal, S. Stewman, S. Konda, P.D. Larkey, and W.G. Wagner, May 1990.

Software Engineering Institute Technical Report CMU/SEI-92-TR-30, Carnegie Mellon University, *Software Development Risk: Opportunity, Not Problem*, by Roger L. Van Scoy, September 1992.

Software Engineering Institute Technical Report CMU/SEI-93-TR-6, Carnegie Mellon University, *Taxonomy Based Risk Identification*, by Marvin J. Carr, Suresh L. Konda, Ira Monarch, Carol Ulrich, and Clay F. Walker, June 1993.

Software Engineering Institute Technical Report CMU/SEI-94-TR-19, Carnegie Mellon University, *Software Risk Evaluation Method Version 1.0*, by Frank J. Sisti and Sujoe Joseph, December 1994.

Software Engineering Institute Technical Review '92, Carnegie Mellon University, *Software Development Risk Management: An SEI Appraisal*, by Robert J. Kirpatrick, Julie A. Walker, and Robert Firth, 1992.

Stormont, D.P. and Welgan, R., "Risk Management for the B-1B Computer Upgrade," *IEEE Proceedings of the IEEE 1994 National Aerospace and Electronics Conference*, vol. 2, pp. 1143-1149, New York, NY, 1994.

Turn, R., Davis, M.R., and Reinstedt, R.N., *A Management Approach to the Development of Computer Based Systems*, The Rand Corporation, 1976.

United States Air Force. "Software Risk Abatement," *Air Force Systems Command Pamphlet 800-45*, 30 September 1988.

United States Congress, *The Status of the V-22 Tiltrotor Aircraft Program*, Hearing Before the Procurement and Military Nuclear Systems Subcommittee and the Research and Development Subcommittee of the Committee on Armed Services, House of Representatives, One Hundred Second Congress, Second Session, 5 August 1992.

United States General Accounting Office, *Technical Risk Assessment; The Status of Current DOD Efforts*, Report to the Chairman, Committee on Governmental Affairs, United States Senate. April 1986.

United States General Accounting Office, GAO/NSIAD-86-45S-7, DOD Acquisition, *Case Study of the Navy V-22 Osprey Joint Vertical Lift Aircraft Program,* Report to Congressional Requesters. July 1986.

United States General Accounting Office, GAO/NSIAD-88-160, *DOD Acquisition Programs, Status of Selected Systems,* Report to the Chairman, Committee on Armed Services, U.S. Senate. June 1988.

United States General Accounting Office, GAO/IMTEC-90-23, *Meeting the Government's Technology Challenge: Results of GAO Symposium*. February 1990.

United States General Accounting Office, GAO/NSIAD-90-30, *Defense Acquisition Programs, Status of Selected Systems*, Report to the Chairman, Committee on Armed Services, U.S. Senate. December 1989.

United States General Accounting Office, GAO/IMTEC-90-34, *Embedded Computers, Better Focus on This Technology Could Benefit Billion Dollar Weapons Programs.* Report to the Chairman, Legislation and National Security Subcommittee, Committee on Government Operations, House of Representatives. April 1990.

United States General Accounting Office, GAO/NSIAD-91-45, *Naval Aviation, The V-22 Osprey -- Progress and Problems*, Report to the Ranking Minority Member, Committee on Armed Services, House of Representatives. October 1990.

United States General Accounting Office, GAO/IMTEC-92-48, *Embedded Computer Systems, Significant Software Problems on C-17 Must Be Addressed*, Report to the Chairman, Subcommittee on Legislation and National Security, Committee on Government Operations, House of Representatives. May 1992.

United States General Accounting Office, GAO/IMTEC-92-62BR, *Embedded Computer Systems, Defense Does Not Know How Much It Spends on Software*, Briefing Report to the Chairman, Subcommittee on Research and Development, Committee on Armed Services, House of Representatives. July 1992.

United States General Accounting Office, GAO/IMTEC-93-13, *Mission Critical Systems, Defense Attempting to Address Major Challenges*, Report to the Chairman, Subcommittee on Research and Development, Committee on Armed Services, House of Representatives. November 1992.

United States General Accounting Office, GAO/NSIAD-93-15, *Weapons Acquisition, A Rare Opportunity for Lasting Change*, Report from the Comptroller General of the United States. December 1992.

United States General Accounting Office, GAO/NSIAD-94-44, *Navy Aviation, V-22 Development -- Schedule Extended, Performance Reduced, and Costs Increased*, Report to the Chairman, Committee on Armed Services, House of Representatives. January 1994.

V-22 Program Document, *Computer Resources Life Cycle Management Plan*, 25 September 1995.

V-22 Program Policy and Procedure Number 211A, *V-22 Risk Management*, 21 October 1994.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
8725 John J. Kingman Rd., STE 0944
Ft. Belvoir, Virginia 22060-6218

2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Rd.
Monterey, California 93943-5101

3. Director, Training and Education 1
MCCDC, Code C46
1019 Elliot Rd.
Quantico, Virginia 22134-5027

4. Professor David V. Lamm, Code SM/Lt 2
Department of Systems Management
Naval Postgraduate School
Monterey, California 93943-5000

5. Professor Martin J. McCaffrey, Code SM/Mf 5
Department of Systems Management
Naval Postgraduate School
Monterey, California 93943-5000

6. Professor Tarek Abdel-Hamid, Code SM/Ah 4
Department of Systems Management
Naval Postgraduate School
Monterey, California 93943-5000

7. Professor George Prosnik 1
9820 Belvoir Road
Defense Systems Management College
Fort Belvoir, Virginia 22060-5426

8. LTC Jim Huskins 1
9820 Belvoir Road
Defense Systems Management College
Fort Belvoir, Virginia 22060-5426

9. Mr. Fred Huber 1
Manager, Navy Programs
801 North Randolph Street, Suite 405
Arlington, Virginia 22203

10. Ms. Tara Potter Rumsey 1
Resident Affiliate - GTE Government Systems
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213-3890

11. Mr. Dick Murphy 1
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213-3890

12. Dr. Elaine Hall 1
Director, Risk Management & Metrics
Software Program Managers Network
P.O. Box 33445
Indialantic, Florida 32903-3445

13. Major Lloyd Whitworth 2
P.O. Box 392
Boerne, Texas 78006